

**UNIVERSIDAD DE LAS CIENCIAS INFORMÁTICAS**  
**Facultad 9**



**TÍTULO:** Herramienta de Apoyo a la Gestión de Proyectos.

**TRABAJO DE DIPLOMA PARA OPTAR POR EL TÍTULO DE INGENIERO EN  
INFORMÁTICA**

**AUTOR(ES):** Jesse Daniel Cano Otero.

**TUTOR(ES):** Lic. José Ángel Lago Graverán

**CO-TUTOR(ES):** Ing. Alexey Díaz Domínguez

**Ciudad de la Habana, 5 de julio del 2008.**

**Año 50 de la Revolución.**

## **DEDICATORIA**

A mi familia por hacerme quién soy.  
A mis amigos por ayudarme a llegar hasta aquí.  
A la flaca por todo el amor.

## **AGRADECIMIENTOS**

A Papun, mi tutor, por la idea del trabajo y por guiarme todo este tiempo.

A Alejo, porque sin su apoyo no hubiese podido completar el trabajo.

A todos los que de una manera u otra hicieron posible el desarrollo del trabajo.

## DECLARACIÓN DE AUTORÍA

Declaro que soy el único autor de este trabajo y autorizo a la Universidad de las Ciencias Informáticas a hacer uso del mismo en su beneficio.

Para que así conste firmo la presente a los \_\_\_\_ días del mes de \_\_\_\_\_ del año \_\_\_\_\_.

Jesse Daniel Cano Otero

José Ángel Lago Graverán

---

## **OPINIÓN DEL TUTOR**

El tutor del presente Trabajo de Diploma considera que durante su ejecución el estudiante mostró las cualidades que a continuación se detallan:

Asumió con un alto grado de responsabilidad todas las tareas trazadas. El trabajo fue desarrollado en condiciones desfavorables, lo cual propició que el autor demostrara un alto grado de independencia, dedicación, seriedad, rigurosidad y dominio del tema estudiado para hacer posible la culminación de esta investigación. De forma general el Trabajo de Diploma presenta una alta calidad científico-técnica. Los resultados obtenidos presentan un elevado nivel de rigurosidad y formalismo, lo cual corrobora que el objetivo propuesto al inicio de la investigación fue alcanzado.

Por todo lo anteriormente expresado se considera que el estudiante está apto para ejercer como Ingeniero en Ciencias Informáticas, y se propone la calificación máxima de 5 puntos.

---

José Angel Lago Graverán

## RESUMEN

La combinación de los conocimientos, habilidades, herramientas y técnicas para obtener resultados exitosos en la ejecución de un proyecto es una tarea complicada. El desarrollo de las tecnologías de la información y las comunicaciones permite apoyar de manera significativa a la Gestión de Proyectos.

Informatizar un grupo importante de procesos de la Gestión de Proyectos puede ser de gran ayuda para cualquier equipo de proyecto interesado en obtener los resultados dentro de los márgenes de tiempo, costo y calidad previstos.

En el presente trabajo se propone iniciar el desarrollo de una herramienta en software libre que informatice la mayoría de los procesos de la Gestión de Proyectos.

Para el desarrollo de la herramienta se utiliza RUP como metodología, UML como lenguaje de modelado, PHP como lenguaje de programación, postgresSQL como gestor de bases de datos, Visual Paradigm como herramienta CASE, ZendStudio como framework de desarrollo y pgAdmin como aplicación gráfica para la gestión de bases de datos en postgresSQL.

## **PALABRAS CLAVES**

Gestión de Proyectos, Herramienta de Apoyo, Software Libre.

## TABLAS Y FIGURAS

Tabla 1: Actores del Negocio.....	52
Tabla 2: Trabajadores del Negocio.....	53
Tabla 3: Actores del Sistema.....	65
Tabla 4: Descripción del Caso de Uso Autenticar Usuario.....	67
Tabla 5: Descripción del CUS Gestionar Documentos.....	73
Tabla 6: Descripción del CUS Manejar Actividades.....	82
Tabla 7: Descripción del CUS Manejar Costos.....	85
Tabla 8: Descripción del CUS Manejar Riesgos.....	89

Figura 1: Diagrama de Casos de uso del Negocio.....	55
Figura 2: Diagrama de Actividades CUN Gestionar el Alcance.....	56
Figura 3: Diagrama de Actividades CUN Gestionar el Tiempo.....	57
Figura 4: Diagrama de Actividades CUN Gestionar el Costo.....	58
Figura 5: Diagrama de Actividades CUN Gestionar los Riesgos.....	58
Figura 6: Diagrama de Actividades CUN Gestionar los RRHH.....	59
Figura 7: Diagrama de Actividades CUN Gestionar la Calidad.....	59
Figura 8: Diagrama de Actividades CUN Gestionar las Comunicaciones.....	60
Figura 9: Diagrama de Actividades CUN Gestionar el Aprovisionamiento.....	60
Figura 10: Diagrama de Casos de Uso Subsistema Autenticación.....	66
Figura 11: Diagrama de Casos de Uso Subsistema Gestor de Documentos.....	67
Figura 12: Diagrama de Casos de Uso Subsistema Procesos.....	74
Figura 13: Diagrama de Clases del Análisis CU Autenticar Usuario.....	91
Figura 14: Diagrama de Clases del Análisis CU Gestionar Documentos.....	91
Figura 15: Diagrama de Clases del Análisis CU Manejar Actividades.....	92
Figura 16: Diagrama de Clases del Análisis CU Manejar Costos.....	92
Figura 17: Diagrama de Clases del Análisis CU Manejar Riesgos.....	92
Figura 18: Diagrama de Clases del Diseño CU Autenticar Usuario.....	93
Figura 19: Diagrama de Clases del Diseño CU Gestionar Documentos.....	94
Figura 20: Diagrama de Clases del Diseño CU Manejar Actividades.....	94
Figura 21: Diagrama de Clases del Diseño CU Manejar Costos.....	95
Figura 22: Diagrama de Clases del Diseño CU Manejar Riesgos.....	95
Figura 23: Diagrama de Clases Persistentes.....	96
Figura 24: Modelo de Datos.....	97



## INDICE

INTRODUCCIÓN .....	1
CAPÍTULO 1: Fundamentación Teórica.....	5
1.1 Introducción .....	5
1.2 Conceptos asociados al dominio del problema.....	5
1.2.1 Project Management Institute (PMI).....	5
1.2.2 PMBOK .....	5
1.2.3 Gestión .....	6
1.2.4 Proyecto .....	6
1.2.5 Gestión de Proyectos. ....	8
1.2.6 Áreas de Conocimiento de la Gestión de Proyectos.....	9
1.3 Objeto de Estudio .....	11
1.3.1 Descripción General .....	11
1.3.2 Áreas de Conocimientos .....	12
1.3.2.1 Gestión del Tiempo .....	12
1.3.2.2 Gestión del Costo .....	13
1.3.2.3 Gestión de los Riesgos.....	14
1.3.2.4 Gestión del Alcance.....	16
1.3.2.5 Gestión de la Calidad.....	17
1.3.2.6 Gestión de los Recursos Humanos .....	19
1.3.2.7 Gestión de las Comunicaciones .....	21
1.3.2.8 Gestión del Aprovisionamiento.....	21
1.3.3 Situación Problemática.....	23
1.4 Estado del Arte.....	23
1.4.1 Herramientas de Apoyo a la Gestión .....	23
1.4.1.1 Microsoft Project.....	23
1.4.1.2 Idinet .....	24
1.4.1.3 GanttPV.....	24
1.4.1.4 dotProject .....	25
1.4.2 ERPs .....	25
1.4.2.1 SAP ERP .....	25
1.4.2.2 ERP Primavera .....	26
1.4.2.3 PeopleSoft .....	27
1.4.2.4 Microsoft Dynamics NAV.....	27
1.4.2.5 OpenPro.....	28
1.5 Conclusiones .....	28
CAPÍTULO 2: Tendencias y tecnologías actuales a desarrollar. ....	29
2.1 Introducción .....	29
2.2 Metodologías de Desarrollo de Software .....	29
2.2.1 Microsoft Solution Framework (MSF) .....	29
2.2.2 Extreme Programming (XP).....	31
2.2.3 Rational Unified Process (RUP) .....	32
2.2.4 Fundamentación de la metodología seleccionada (RUP).....	35
2.3 Lenguaje de Modelación: Unified Modeling Language (UML) .....	35
2.4 Arquitectura de Software.....	37
2.4.1 Modelo Vista Controlador.....	37
2.5 Lenguajes de Desarrollo Web .....	38
2.5.1 ASP .....	38
2.5.2 ASP.NET .....	38
2.5.3 PERL .....	39
2.5.4 JSP .....	40
2.5.5 PHP .....	41
2.5.6 Fundamentación del lenguaje seleccionado (PHP) .....	41

2.6 Sistemas Gestores de Bases de Datos .....	42
2.6.1 SQL Server 2000.....	42
2.6.2 MySQL .....	43
2.6.3 Oracle.....	44
2.6.4 PostgreSQL .....	45
2.6.5 Fundamentación del Sistema Gestor Seleccionado.....	46
2.7 Herramientas Propuestas para el Desarrollo de la Aplicación.....	47
2.7.1 Visual Paradigm .....	47
2.7.2 Zend Studio .....	49
2.7.3 PgAdmin III .....	50
2.8 Conclusiones .....	51
Capítulo 3 “Presentación de la solución propuesta” .....	52
3.1 Introducción .....	52
3.2 Modelo de Negocio .....	52
3.2.1 Actores y trabajadores del negocio.....	52
3.2.2 Procesos de Negocio.....	53
3.2.3 Diagrama de Casos de Uso del Negocio .....	55
3.2.4 Diagramas de Actividades de los Casos de Uso de Negocio .....	56
3.3 Requerimientos del Software.....	61
3.3.1 Requerimientos Funcionales.....	61
3.3.2 Requerimientos No Funcionales.....	63
3.4 Descripción del Sistema Propuesto.....	64
3.4.1 Descripción de los actores.....	64
3.4.2 Casos de Uso del Sistema.....	66
3.4.2.1 Subsistema Autenticación.....	66
3.4.2.2 Subsistema Gestor de Documentos.....	67
3.4.2.3 Subsistema Procesos.....	74
3.5 Conclusiones.....	90
CAPÍTULO 4: Construcción de la solución propuesta.....	91
4.1 Introducción .....	91
4.2 Modelo del Análisis.....	91
4.2.1 Diagrama de Clases del Análisis.....	91
4.2.1.1 Subsistema Autenticación.....	91
4.2.1.2 Subsistema Gestor de Documentos.....	91
4.2.1.3 Subsistema Procesos.....	92
4.3 Modelo del Diseño.....	93
4.3.1 Diagramas de Clases del Diseño.....	93
4.3.1.1 Subsistema Autenticación.....	93
4.3.1.2 Subsistema Gestor de Documentos.....	94
4.3.1.3 Subsistema Procesos.....	94
4.4 Modelo de Datos.....	96
4.5 Conclusiones.....	97
Conclusiones .....	98
Recomendaciones .....	99
Bibliografía Consultada.....	100
Citas Bibliográficas .....	102
Glosario de Siglas y Términos .....	103

# INTRODUCCIÓN

El rápido desarrollo de las tecnologías de la información y las comunicaciones (TICs) ha provocado su difusión a nivel mundial, la Asociación Americana de las Tecnologías de la Información (ITAA por sus siglas en inglés) define a las TICs como el estudio, diseño, desarrollo, implementación, soporte o administración de sistemas de información basados en computadoras, básicamente es el uso de computadoras y software para convertir, almacenar, proteger, procesar, transmitir y recuperar de manera segura la información.

Sin embargo, este proceso de difusión de las TICs en el mundo es algo muy complejo debido a que las grandes transnacionales de las potencias mundiales distribuyen las TICs de acuerdo a sus intereses políticos y económicos, es por eso que un país como Cuba tiene que hacer un esfuerzo sobrenatural para lograr informatizar su sociedad mediante el uso de estas tecnologías.

Un ejemplo claro de este esfuerzo que realiza nuestro país, es la Universidad de las Ciencias Informáticas (UCI) que desde sus inicios comenzó a desarrollar software para informatizar procesos, algunos internos de la universidad, otros referentes a varias áreas de la sociedad y otros para empresas extranjeras. En la actualidad nuestra universidad tiene varios proyectos de desarrollo de software para aplicarse en la hermana República Bolivariana de Venezuela y específicamente con PDVSA, donde, como es la tendencia actual a nivel mundial la mayoría de sus operaciones se ejecutan a través de proyectos.

A la hora de gestionar un proyecto, dígame un proyecto de cualquier índole, hay que tener muchos factores en cuenta, hay un grupo de conocimientos, habilidades, herramientas y técnicas que deben ser combinados correctamente para cumplir los requisitos de dicho proyecto. El Project Management Institute (PMI) ha generado un libro donde agrupa el conocimiento de como combinar efectivamente los factores a tener en cuenta a la hora de gestionar un proyecto.

La gestión de proyectos es una tarea muy difícil si tenemos en cuenta la gran cantidad de procesos que hay que controlar, la gran cantidad de variables que hay que tener en cuenta a la hora de tomar una decisión, en fin, es un trabajo que se puede tornar muy engorroso e

incluso inefectivo si no se tiene la debida organización del trabajo, y aun con un grado elevado de organización existe el riesgo del error humano.

El desarrollo de las TICs nos permite informatizar algunos procesos de la gestión de proyectos, con el fin de organizar el trabajo así como asegurar la disponibilidad de la información y facilitar el proceso de toma de decisiones.

Actualmente existen algunas herramientas que de una manera u otra han informatizado algunos procesos de la gestión de proyectos como lo son MsProject, Idinet, GanttPV (herramientas de apoyo a la gestión de proyectos) u otros que en realidad son paquetes de aplicaciones que informatizan un rango más amplio de procesos, hasta llegar a niveles de complejidad elevados como lo son SAP, Primavera y Navision (ERPs).

El uso de un ERP o una herramienta de apoyo reporta numerosos beneficios para la organización: se reducen el tiempo de ejecución y los costos de los procesos claves del negocio, se logra que las transacciones de la información sean más rápidas, se logra una mejor administración financiera, se logra mejorar la calidad y con ello la satisfacción del cliente, ofrecen la oportunidad de medir los resultados de una manera continua. **(1)**

Múltiples estudios muestran que la implantación de alguna herramienta de apoyo a la gestión o algún sistema ERP es hoy en día una tarea costosa, principalmente para las organizaciones pequeñas y medianas (SMEs por sus siglas en inglés), esto provoca que muchas SMEs no tengan los recursos suficientes, debido al gasto tan grande de la implantación. **(2)**

Para evitar estos altos costos, sería conveniente el uso de una herramienta de apoyo o un ERP desarrollado en software libre, para, de este modo, evitar el gasto de recursos en temas de licenciamiento del software, sin embargo, cualquiera que realice una búsqueda de un software con estas características llegará a la conclusión de que, en software libre, existen herramientas que informatizan solo algunos de los procesos de la gestión de proyectos.

La situación problemática viene dada debido a:

- La compleja aplicación de los conocimientos generados por la organización líder en el tema (PMI).

- El alto costo en tiempo y recursos que acarrea la gestión de proyectos sin la ayuda de las TICs.
- El alto costo de las herramientas propietarias de apoyo a la gestión de proyectos.
- La mayoría de las herramientas software libre no abarcan completamente un grupo de procesos de la gestión de proyectos como los de la Gestión de la Calidad, del Aprovisionamiento y los Riesgos.

Por lo tanto el presente trabajo consiste en diseñar, en software libre, una herramienta que:

- Sirva de apoyo a la Gestión de Proyectos independientemente del área social donde se desarrolle.
- Cuenten con documentación suficiente que ayude a entender los procesos de la Gestión de Proyectos de manera general y así facilitar su aplicación.
- Que sea capaz de brindar una solución que mejore sustancialmente el proceso de toma de decisiones en cualquier proyecto.

El **problema científico** es la falta de herramientas en software libre que brinden una solución que abarque a fondo todos los procesos de la Gestión de Proyectos, específicamente los relacionados con la Gestión de la Calidad, el Aprovisionamiento y los Riesgos.

El **objeto de estudio** del presente trabajo son las “Buenas Prácticas” de la Gestión de Proyectos según las teorías del Project Management Institute.

Este trabajo tendrá como **campo de acción** las distintas Áreas de Conocimientos de la Gestión de Proyectos que dictan como realizar una buena práctica a la hora de gestionar un proyecto, según la organización líder en el tema, específicamente las Áreas de Gestión de la Calidad, el Aprovisionamiento y los Riesgos, así como los procesos que se incluyen en estas.

El **objetivo general** de este trabajo es iniciar el desarrollo en software libre de una herramienta de apoyo a la gestión de proyectos, que brinde a los entes ejecutores una vía de optimizar los procesos de control y toma de decisiones y que además tenga en cuenta los procesos involucrados en la Gestión de la Calidad, el Aprovisionamiento y los Riesgos.

Las tareas a ejecutar para el cumplimiento del objetivo del trabajo serian:

1. Estudio del Estado del arte
2. Estudio de las mejores prácticas en la gestión de proyectos según la organización líder.
3. Investigar las tendencias tecnológicas actuales y seleccionar las herramientas a utilizar para el desarrollo de la aplicación.
4. Elaborar una solución al problema científico.
5. Realizar el análisis y diseño de la herramienta propuesta en la solución.

### **Hipótesis:**

Si se desarrolla una herramienta como la que se propone quedarán informatizados gran parte de los procesos de la gestión de proyectos, incluyendo los relacionados con la Gestión de la Calidad, el Aprovisionamiento y los Riesgos dando como resultado una aplicación en software libre que será de gran utilidad para todo proyecto en ejecución.

### **Métodos de Investigación**

#### 1. Métodos Teórico

##### 1.1. Lógico.

1.1.1. Modelación: para crear un modelo (abstracción) que explique el resultado final del trabajo.

1.2. Histórico: para el estudio de antecedentes que existen respecto a de herramientas de este tipo y programas o software que implementen los diferentes procesos de la gestión de proyectos.

#### 2. Métodos Empíricos

2.1. Revisión de documentos para apoyar el estudio de los diferentes procesos de la gestión de proyectos.

# **CAPÍTULO 1: Fundamentación Teórica.**

## ***1.1 Introducción***

A nivel mundial, actualmente, existen disímiles tipos de proyectos, cada uno con gran cantidad de particularidades, a simple vista es fácil percatarse, entonces, que el método para gestionar dichos proyectos es algo sumamente complicado. Para el desarrollo de una herramienta que apoye la Gestión de Proyectos es necesario, entonces, apropiarse del fundamento teórico de dicha práctica con el fin de ser objetivos en el desarrollo de la solución.

En este capítulo se profundiza en los conceptos esenciales relacionados con la Gestión de Proyectos, desde la definición de lo que esta práctica significa hasta los conceptos establecidos por la institución rectora (Project Management Institute), así como un estudio de las soluciones existentes que incluye las funcionalidades que estas brindan, la argumentación de las diferencias que existen entre dichas soluciones y el objetivo a lograr con el desarrollo de la herramienta que se propone en este trabajo.

## ***1.2 Conceptos asociados al dominio del problema***

### **1.2.1 Project Management Institute (PMI)**

Con mas de 260 000 miembros distribuidos en 171 países, PMI es la organización líder para la profesión que es la Gestión de Proyectos. PMI está comprometido con la defensa activa de la profesión, estableciendo estándares profesionales (el más reconocido es el PMBOK), conduciendo investigaciones y proveyendo recursos. El PMI además, promociona el desarrollo profesional y brinda certificaciones y las oportunidades del trabajo en comunidad. **(3)**

Por más de 40 años PMI ha impulsado la carrera de los profesionales que hacen la Gestión de Proyectos indispensable para la consecución de resultados en un negocio.

### **1.2.2 PMBOK**

El libro “A Guide to the Project Management Body of Knowledge” más conocido como PMBOK es el estándar más ampliamente reconocido para gestionar y administrar proyectos. **(4)** Resulta curioso que este texto tenga la fama de ser un manual para dirigir proyectos, o bien que se trata de un texto rigorista y dogmático. Lo cierto es que se trata de una obra realizada

por personas con un agudo sentido práctico, y que tiene incorporada la concepción de que un proyecto exitoso va a ser resultado de la colaboración de todos los miembros del equipo de proyecto.

### **1.2.3 Gestión**

En términos generales los conceptos de administración, gerencia y gestión, son sinónimos a pesar de los grandes esfuerzos y discusiones por diferenciarlos. En la práctica se observa que el término *management* es traducido como administración pero también como gerencia y gestión. En algunos países la administración esta mas referida a lo público y la gerencia a lo privado. **(5)**

Lo esencial de los conceptos administración, gestión y gerencia está en que los tres se refieren a un proceso de “planear, organizar, dirigir, evaluar y controlar”.

La gestión se vería en 2 niveles:

El lineal o tradicional donde es sinónimo de administración: "Por gestión se entiende el conjunto de diligencias que se realizan para desarrollar un proceso o para lograr un producto determinado". Se asume como dirección y gobierno, actividades para hacer que las cosas funcionen, con capacidad para generar procesos de transformación de la realidad.

Con una connotación más actualizada o gerencial la gestión es planteada como "una función institucional global e integradora de todas las fuerzas que conforman una organización". En ese sentido la gestión hace énfasis en la dirección y en el ejercicio del liderazgo.

### **1.2.4 Proyecto**

Un proyecto, en el contexto referente a la gestión de proyectos consiste en un esfuerzo temporal dirigido a la creación de un producto, servicio o resultado. **(4)**

En este contexto, por definición, los proyectos tendrán determinadas características:

#### *1 – Temporales*

Significa que todo proyecto tendrá un comienzo y final bien definidos. El fin del proyecto llega cuando los objetivos de estos han sido alcanzados, cuando es un hecho que dichos objetivos



no podrán cumplirse o cuando la necesidad que dio origen al proyecto ha dejado de existir. El término temporal no implica corta duración, podremos encontrar proyectos que duren varios años. De cualquier manera, sin embargo, la duración de un proyecto es finita.

Es necesario destacar, que normalmente la característica temporal no se aplica al producto, servicio o resultado creado por el proyecto, de hecho muchos de los proyectos se conciben para generar una salida que puede perdurar en el tiempo.

## *2 – Producto, servicio o resultado únicos.*

Un proyecto produce entregables únicos, que pueden ser productos, servicios o resultados.

Un proyecto puede generar:

Un producto o artefacto que es producido y es cuantificable.

Una capacidad para brindar un servicio.

Un resultado, por ejemplo un proyecto de investigación que genera conocimiento.

El ser único, es una característica fundamental de los entregables del proyecto. Por ejemplo, en el mundo existen millones de edificios de oficinas, pero cada uno en particular es único: diferentes dueños, diferentes diseños, diferentes ubicaciones, diferentes contratistas, etc. La presencia de elementos repetitivos no cambia la esencia única del resultado generado por el proyecto.

## *3 – Elaboración Progresiva*

Esta característica de los proyectos acompaña los conceptos de temporal y único. La elaboración progresiva es sinónimo de desarrollo por paso. Por ejemplo, el alcance de un proyecto puede ser descrito superficialmente al comienzo de este, y luego ser descrito de una manera más explícita a medida que el equipo de proyecto profundice su conocimiento acerca de cómo alcanzar los objetivos y la manera de producir los entregables.

Es válido destacar que la elaboración progresiva debe ser coordinada cuidadosamente con una correcta definición del alcance, sobre todo si el proyecto está siendo desarrollado bajo contrato. Cuando el alcance del proyecto esté correctamente definido, debe ser controlado mientras las especificaciones del producto son elaboradas progresivamente.

La diferencia entre un proyecto y una operación viene dada por la continuidad y la repetitividad de las operaciones, mientras que los proyectos llegan a su fin en un intervalo de tiempo. Las operaciones normalmente producen el mismo resultado una y otra vez, los proyectos, en contraste, generan un resultado único.

### **1.2.5 Gestión de Proyectos.**

La gestión de proyectos es la aplicación de conocimientos, habilidades, herramientas y técnicas en las actividades del proyecto para cumplir los requisitos de este. La gestión de proyectos es completada a través de la aplicación e integración de procesos de gestión de proyectos que son: la iniciación, el planeamiento, la ejecución, el control y monitoreo y el cierre. El líder del proyecto es la persona responsable del cumplimiento de los objetivos de este.

Gestionar un proyecto incluye:

- Identificar los requisitos.
- Establecer objetivos claros y alcanzables.
- Balancear las variables de calidad, alcance, tiempo y costo.
- Adaptar las especificaciones, planes y enfoques a las distintas preocupaciones y expectativas de varios stakeholders.

Los líderes de proyectos usualmente hablan de una “triple restricción” al gestionar los requisitos del proyecto: el alcance, duración y costo. La calidad del proyecto es proporcional al balance de estos 3 factores. Los proyectos de alta calidad entregan el producto, servicio o resultado requerido, dentro del alcance, en tiempo y dentro del presupuesto asignado al proyecto. La relación entre estos factores es tan estrecha que si uno de ellos cambia, al

menos uno de los otros probablemente se vea afectado. Los líderes de proyectos deben gestionar también las respuestas a lo incierto. Un riesgo en un proyecto es un evento o condición inciertos que, si ocurre, conlleva un efecto, negativo o positivo, en al menos un objetivo del proyecto. **(4)**

El equipo de proyecto tiene una responsabilidad profesional con los stakeholders, los clientes, la organización ejecutora y el público. Los miembros del PMI se adhieren obligatoriamente a un “Código de Ética” y aquellos con la certificación “Project Management Professional (PMP®)” se adhieren con igual obligatoriedad a un “Código de Conducta Profesional”.

Es importante destacar que muchos de los procesos dentro de la gestión de proyectos son iterativos debido a la existencia y necesidad de la elaboración progresiva.

La terminología “Gestión de Proyectos” es comúnmente usada para describir un enfoque organizacional de la administración de proyectos y algunas operaciones, que pueden ser redefinidas como proyectos. Una organización que adopte este enfoque define sus actividades como proyectos. En los últimos años ha habido una tendencia a administrar cada vez más actividades en cada vez mas áreas de aplicación usando el enfoque de gestión de proyectos. Esto no quiere decir que todas las operaciones puedan o deban ser redefinidas como proyectos.

### **1.2.6 Áreas de Conocimiento de la Gestión de Proyectos**

Las áreas de conocimiento de la gestión de proyectos, agrupan los procesos de la gestión dentro nueve áreas, que se describen brevemente a continuación.

- Gestión del alcance del proyecto.

Describe los procesos involucrados en la determinación de que el proyecto incluye todo el trabajo requerido y solo el trabajo requerido para completarse exitosamente. Consiste en el planeamiento del alcance, la definición del alcance, la creación del WBS (Work Breakdown Structure), la verificación del alcance y el control del alcance de los procesos del proyecto.

- Gestión del tiempo del proyecto.

Describe los procesos referentes a la terminación en tiempo del proyecto. Consiste en la definición de las actividades, la estimación de los recursos por actividades, la estimación de la duración por actividad, el desarrollo del Schedule y el control del Schedule de los procesos del proyecto.

- Gestión del Costo del Proyecto.

Describe los procesos involucrados en el planeamiento, estimación, elaboración del presupuesto y el control de los costos para que el proyecto pueda ser completado dentro del presupuesto general aprobado para el. Consiste en la estimación de los costos, presupuestación de los costos y el control de los costos de los procesos del proyecto.

- Gestión de la Calidad del proyecto.

Describe los procesos involucrados en asegurar que el proyecto satisfaga los objetivos para lo que fue creado. Consiste en el planeamiento de la calidad, el aseguramiento de la calidad y el control de la calidad de los procesos del proyecto.

- Gestión de los Recursos Humanos del proyecto.

Describe los procesos que organizan y gestionan el equipo de proyecto. Consiste en el planeamiento de los recursos humanos, en la adquisición del equipo de proyecto, el desarrollo del equipo de proyecto, y en el control del equipo de proyecto.

- Gestión de las comunicaciones del proyecto.

Describe los procesos concernientes al la generación, recolección, diseminación, archivamiento y disposición adecuadas y en tiempo de la información del proyecto. Consiste en el planeamiento de las comunicaciones, la distribución de la información, el reporte de rendimiento y la gestión de estos con los stakeholders del proyecto.

- Gestión de los riesgos del proyecto.

Describe los procesos concernientes a la conducción de la administración de los riesgos del proyecto. Consiste en el planeamiento de la gestión de los riesgos, la identificación de los riesgos, el análisis cualitativo de los riesgos, el análisis cuantitativo de los riesgos, el planeamiento de las respuestas a los riesgos y el control y monitoreo de los riesgos del proyecto.

- Gestión de aprovisionamiento del proyecto.

Describe los procesos involucrados en la compra o adquisición de productos, servicios o resultados así como en la gestión de los procesos de la administración de los compromisos contractuales. Consiste en el planeamiento de las compras y adquisiciones, el planeamiento de los contratos, la solicitud de disponibilidad a posibles proveedores, la selección de los proveedores, la administración de los contratos y la gestión de los procesos de cierre de contratos del proyecto. **(4)**

### ***1.3 Objeto de Estudio***

#### **1.3.1 Descripción General**

Los proyectos son, en la actualidad más que nunca, un hecho en la vida de la mayoría de las organizaciones. Se esta haciendo cada día mas común que las organizaciones obtengan sus objetivos y aseguren su financiamiento mediante el desarrollo de proyectos específicos. Los inversionistas por su parte piden detalles de como los proyectos serán gestionados.

Ante este ambiente hay una clara necesidad de experiencia en cuanto a Gestión de Proyectos.

Una organización puede ser pequeña o grande, pública o privada, vieja o nueva. Una organización puede usar los proyectos diariamente para garantizar la entrega de productos o servicios a sus clientes, puede usar los proyectos esporádicamente, como cuando se usan para diseñar e instalar sistemas computarizados para implementar nuevos procedimientos. Cualquiera que sea el caso, una organización usa los proyectos para obtener nuevos resultados. Algunas organizaciones lo hacen bien, otras van de un desastre a otro sin aprender de los errores del pasado.

Aplicando los principios y métodos de la Gestión de Proyectos se pueden evitar problemas frecuentes. La Gestión de Proyectos como disciplina es, hoy en día, relativamente madura y puede ser aplicada exitosamente a cualquier tipo de proyecto, pequeño o grande.

Mediante la aplicación de los principios y la buena práctica de la Gestión de Proyectos, es probable que una organización pueda:

- Terminar sus proyectos en tiempo y dentro del presupuesto definido.
- Entregar un producto o servicio con calidad que complazca las expectativas del cliente y los usuarios.
- Asegurarse de que el equipo de proyecto terminará su labor con un sentimiento de logro y satisfacción personal.

Es por esto que en el presente trabajo se pretende diseñar una herramienta bajo los principios de la Gestión de Proyectos según la institución líder, incluyendo todos los procesos de la disciplina, en especial lo referentes a la Gestión de la Calidad, el Aprovechamiento y los Riesgos, para facilitar el trabajo y el proceso de toma de decisiones de aquellos que pretenden gestionar sus proyectos con efectividad.

### **1.3.2 Áreas de Conocimientos**

Los procesos de cada área de conocimiento interactúan entre sí y con otros procesos de otras áreas de conocimiento. Cada proceso implica el esfuerzo de una o varias personas o grupos de personas, basándose en las necesidades del proyecto. Cada proceso ocurre al menos una vez en cada proyecto y si este está dividido en fases cada proceso ocurre en una o más fases. Es necesario destacar que los procesos se describen aquí como componentes discretos con campos de acción bien definidos pero en la práctica estos se pueden solapar e interactuar de formas imposibles de definir. **(4)**

#### **1.3.2.1 Gestión del Tiempo**

La gestión del tiempo del proyecto incluye los procesos requeridos para el cumplimiento en tiempo de este. La gestión del tiempo del proyecto incluye lo siguiente:

*Definición de las Actividades:* Consiste en la identificación de las actividades específicas que necesitan ser desarrolladas para generar varios de los entregables del proyecto.

*Secuenciamiento de las Actividades:* Consiste en identificar y documentar la dependencia entre las actividades del proyecto.

*Estimación de los recursos de las actividades:* Consiste en estimar el tipo y las cantidades de recursos necesario para ejecutar cada actividad del proyecto.

*Estimación de la duración de las actividades:* Consiste en estimar la cantidad de períodos de trabajo necesarios para completar cada actividad del proyecto.

*Desarrollo del Schedule:* Consiste en analizar la secuencia de las actividades, las duraciones, los requerimientos de recursos y las limitaciones del Schedule para crear el Schedule del proyecto.

*Control del Schedule:* Consiste en controlar los cambios del Schedule del proyecto.

En algunos proyectos, especialmente aquellos de corto alcance, el secuenciamiento de las actividades, la estimación de los recursos de las actividades, la estimación de la duración y el desarrollo del Schedule están tan estrechamente vinculados que son vistos como un solo proceso que puede ser ejecutado por una persona en un período de tiempo relativamente corto. **(4)**

### **1.3.2.2 Gestión del Costo**

La gestión del costo del proyecto incluye los procesos involucrados en el planeamiento, estimación, presupuestación y control de los costos para que el proyecto pueda ser terminado dentro del presupuesto general aprobado. Esta área de conocimiento tiene tres procesos:

*Estimación de Costos:* Consiste en desarrollar una aproximación de los costos de los recursos necesarios para completar cada actividad del proyecto.

*Presupuestación de los Costos:* Consiste en balancear los costos estimados de las actividades individuales o paquetes de trabajo para establecer una línea base de los costos.

*Control de los Costos:* Consiste en ejercer influencia sobre los factores que generan variaciones en los costos y controlar los cambios del presupuesto del proyecto.

La gestión del costo del proyecto se preocupa en primer lugar del costo de los recursos necesarios para llevar a cabo las actividades definidas en el Schedule del proyecto. Sin embargo, la gestión del costo debe tener en cuenta además el efecto de las decisiones del proyecto sobre costo del uso, mantenimiento y soporte del producto, servicio o resultado del proyecto. Por ejemplo, limitar el número de revisiones del diseño puede reducir el costo del proyecto a expensas de un incremento en el costo de operación del cliente.

Este punto de vista amplísimo de lo que es la gestión del costo del proyecto es conocida también como costeo del ciclo de vida. El costeo del ciclo de vida, junto con el valor de las técnicas ingenieriles puede mejorar el proceso de toma de decisiones y son usados para reducir el costo, el tiempo de ejecución y para mejorar la calidad y rendimiento de los entregables del proyecto.

La gestión del costo del proyecto tiene en cuenta los requisitos de información de los stakeholders. Los distintos stakeholders medirán los costos del proyecto en distintas formas y en momentos diferentes.

En algunos proyectos, en especial los de corto alcance, la estimación y presupuestación de los costos están relacionados tan estrechamente que pueden ser vistos como un solo proceso y pueden ser ejecutados por una sola persona en un rango de tiempo relativamente corto. Estos procesos se describen como procesos distintos porque las herramientas y técnicas para ejecutarlos son distintas. **(4)**

### **1.3.2.3 Gestión de los Riesgos**

La gestión de los riesgos del proyecto incluye los procesos concernientes con la conducción del planeamiento de la gestión, identificación, análisis, respuestas, monitoreo y control de los riesgos del proyecto, muchos de estos procesos son actualizados durante el proyecto. Los objetivos de la gestión de los riesgos del proyecto son incrementar la probabilidad y el impacto de los eventos positivos y disminuir la probabilidad y el impacto de los eventos adversos al proyecto. Los procesos de la gestión de los riesgos del proyecto son:



*Planeamiento de la gestión de los riesgos:* Consiste en decidir como enfocar, planear y ejecutar las actividades de la gestión de los riesgos del proyecto.

*Identificación de los riesgos:* Consiste en determinar cuales riesgos pueden afectar al proyecto y documentar sus características.

*Análisis cualitativo de los riesgos:* Consiste en priorizar los riesgos para su futuro análisis tasando y combinando su probabilidad de ocurrencia e impacto.

*Análisis cuantitativo de los riesgos:* Consiste en analizar numéricamente el efecto general sobre los objetivos del proyecto que tendrían, de ocurrir, los riesgos identificados.

*Planeamiento de la respuestas a los riesgos:* Consiste en desarrollar opciones y acciones para mejorar oportunidades y reducir las amenazas a los objetivos del proyecto.

*Monitoreo y Control de los Riesgos:* Consiste en seguir los riesgos identificados, monitorear los riesgos residuales, identificar nuevos riesgos, ejecutar los planes de respuesta a los riesgos y evaluar su efectividad a través del ciclo de vida del proyecto.

Un riesgo del proyecto es un evento o condición que, si ocurre, tiene un efecto positivo o negativo en al menos un objetivo del proyecto, ya sea en tiempo, costo, alcance o calidad. Un riesgo puede tener una o mas causas y, de ocurrir, una o mas consecuencias. Por ejemplo, una causa podría ser la necesidad de un permiso de una agencia ambiental para hacer un trabajo, o tener poco personal asignado al diseño del proyecto. El evento del riesgo sería que la agencia ambiental que debe emitir el permiso se tome más tiempo del planeado para emitirlo, o que el personal de diseño asignado no sea adecuado para la actividad. Si cualquiera de estos dos eventos inciertos ocurriera, podría haber un impacto en el costo, Schedule o rendimiento del proyecto. Las condiciones de los riesgos pueden incluir aspectos como el ambiente del proyecto o la organización ejecutora que contribuyan a la ocurrencia del riesgo, por ejemplo: mala práctica de la gestión de proyectos, falta de sistemas integrados de gestión o dependencia de participantes externos que no pueden ser controlados.

Los riesgos del proyecto tienen sus orígenes en lo incierto que hay en todo proyecto. Los riesgos conocidos son aquellos que han sido identificados y analizados, así como que ya sea posible prepararse para estos riesgos usando los procesos descritos en este epígrafe. Los riesgos desconocidos no pueden ser gestionados activamente y una respuesta prudente sería planificar contingencias generales en contra de estos, así como en contra de cualquier riesgo conocido para el que no exista la posibilidad de desarrollar una respuesta activa **(4)**.

Las personas y, por extensión, las organizaciones tienen actitudes hacia los riesgos que afectan la efectividad de la percepción del riesgo y la manera de responder ante él. Para cada proyecto debe ser desarrollado un enfoque consistente hacia los riesgos y que cumpla con los requisitos de la organización, la comunicación acerca de los riesgos, así como el manejo de estos, debe efectuarse de manera abierta y honesta.

Para ser exitosa, la organización debe estar decidida a gestionar los riesgos activa y consistentemente durante todo el ciclo de vida del proyecto.

#### **1.3.2.4 Gestión del Alcance**

La gestión del alcance del proyecto incluye los procesos requeridos para asegurar que el proyecto ejecute todo el trabajo necesitado y solo el trabajo necesitado para completarse exitosamente. La gestión del alcance del proyecto se preocupa en primer lugar por la definición y el control de lo que se incluye y lo que no se incluye en el proyecto. Esta área de conocimiento consta de los siguientes procesos:

*Definición del Alcance:* Consiste en desarrollar una declaración detallada del alcance del proyecto como bases para decisiones futuras del proyecto.

*Creación del WBS:* Consiste en subdividir los mayores entregables del proyecto en componentes mas pequeños y manejables.

*Verificación del Alcance:* Consiste en formalizar la aceptación de los entregables del proyecto que sean completados.

*Control del Alcance:* Consiste en controlar los cambios del alcance del proyecto.

En el contexto del proyecto, el término alcance se refiere a:

- Alcance del Producto. Las características y funciones que caracterizan al producto, servicio o resultado.
- Alcance del Proyecto. El trabajo que necesita ser ejecutado para entregar el producto, servicio o resultado con las características y funciones previamente especificadas.

Los procesos de la gestión del alcance del proyecto, sus herramientas y técnicas, varían según el área de aplicación, estos, usualmente, son definidos como parte del ciclo de vida del proyecto, y son documentados en el plan de gestión del alcance del proyecto. La declaración detallada del alcance del proyecto que fue aprobada junto con el WBS y el Diccionario del WBS son la línea base del alcance del proyecto.

Un proyecto generalmente resulta en un solo producto, pero ese producto puede incluir componentes secundarios, cada uno con su propio separado, pero independiente, alcance de producto. Por ejemplo, un nuevo teléfono incluirá generalmente cuatro componentes secundarios: hardware, software, entrenamiento e implementación.

El completamiento del alcance del proyecto es medido contra el plan de gestión del alcance del proyecto, la declaración del alcance del proyecto, el WBS y el diccionario del WBS, pero el completamiento del alcance del producto se mide contra los requisitos del producto. La gestión del alcance del proyecto necesita ser integrada correctamente con los procesos de las demás áreas de conocimientos, para que el trabajo del proyecto termine con la entrega del producto con el alcance especificado. **(4)**

#### **1.3.2.5 Gestión de la Calidad**

Los procesos de la gestión de la calidad del proyecto incluyen todas las actividades de la organización ejecutora que determinan las políticas de calidad, objetivos y responsabilidades para que el proyecto satisfaga las necesidades para lo cual fue concebido. La gestión de la calidad del proyecto implementa el sistema de gestión de la calidad a través de políticas, procedimientos, y procesos de planeamiento de la calidad, aseguramiento de la calidad y control de la calidad con un desarrollo continuo de las actividades de los procesos. Los procesos de la gestión de la calidad son los siguientes:

*Planeamiento de la Calidad:* Consiste en identificar cuales son los estándares de calidad relevantes al proyecto y determinar como satisfacerlos.

*Aseguramiento de la Calidad:* Consiste en aplicar lo planeado, actividades sistemáticas para asegurar que el proyecto emplea todos los procesos necesitados para cumplir los requisitos.

*Control de la Calidad:* Consiste en monitorear los resultados específicos del proyecto para determinar si estos obedecen los estándares de calidad e identificar las vías para eliminar las causas del rendimiento no satisfactorio.

El enfoque básico de la gestión de la calidad descrito en este epígrafe tiene la intención de ser compatible con la Organización Internacional para la Estandarización (ISO por sus siglas en ingles).

La gestión de la calidad del proyecto debe dirigir la gestión del proyecto y del producto del proyecto. Mientras la gestión de la calidad del proyecto se aplica a todos los proyectos, independientemente de la naturaleza de sus productos, las medidas y técnicas de la calidad del producto son específicas del tipo particular de producto producido por el proyecto. Por ejemplo, la gestión de la calidad de productos de software vincula diferentes enfoques y medidas que las que vincularía la gestión de la calidad de la construcción de una planta nuclear, mientras que la gestión de la calidad del proyecto esta enfocada para servirle a ambos. En cualquier caso, el fracaso a la hora de cumplir los requisitos de calidad en cualquier dimensión puede tener serias consecuencias adversas para alguno o todos los stakeholders. Por ejemplo:

- Cumplir los requisitos del cliente trabajando en demasía puede producir consecuencias negativas como lo son el incremento de las fricciones entre empleados, errores infundados o algún trabajo hecho varias veces.
- Cumplir los objetivos del Schedule del proyecto apurando las inspecciones de calidad planificadas puede producir consecuencias negativas cuando los errores pasen sin ser detectados por dichas inspecciones.

La calidad es el grado con el que el grupo de características inherentes cumplen los requisitos. Las necesidades expuestas e implicadas son la entrada del desarrollo de los requisitos del proyecto. Un elemento crítico en la gestión de la calidad en el contexto del proyecto es convertir las necesidades, deseos y expectativas de los stakeholders en requisitos. **(4)**

La gestión moderna de la calidad complementa la gestión de proyectos. Por ejemplo, ambas disciplinas reconocen la importancia de:

- La satisfacción del cliente. Entender, evaluar, definir y gestionar las expectativas para que los requisitos del cliente sean cumplidos. Esto requiere la combinación de la conformidad con los requisitos (el producto debe producir lo que se dijo que iba a producir) y la aptitud para el uso (el producto o servicio debe satisfacer necesidades reales).
- La prevención, más importante que la inspección. El costo de prevenir errores es generalmente mucho menor que el costo de corregirlos cuando son descubiertos por alguna inspección.
- Responsabilidad de la administración. El éxito requiere la participación de todos los miembros del equipo, pero siempre será responsabilidad de la administración el proveer los recursos necesarios para obtenerlo.
- Desarrollo Continuo. El ciclo de planificar-ejecutar-comprobar-actuar es la base para el mejoramiento de la calidad. Además, las iniciativas en el mejoramiento de la calidad adoptadas por la organización ejecutor, como TQM y Six Sigma, pueden mejorar la calidad de la gestión del proyecto así como la calidad del producto del proyecto.

#### **1.3.2.6 Gestión de los Recursos Humanos**

La gestión de los recursos humanos del proyecto incluye los procesos de organizar y gestionar el equipo de proyecto. El equipo de proyecto son las personas que tienen roles y responsabilidades asignadas para completar el proyecto. Mientras es muy común oír hablar de roles y responsabilidades siendo asignadas, los miembros del equipo deberían estar involucrados en gran parte del planeamiento del proyecto y en el proceso de toma de

decisiones. El tipo y número de los miembros del equipo de proyecto puede cambiar frecuentemente mientras avanza el proyecto. La gestión de los recursos humanos del proyecto incluye procesos como:

*Planeamiento de los Recursos Humanos:* Consiste en identificar y documentar los roles, responsabilidades y relaciones del proyecto.

*Adquisición del equipo de proyecto:* Consiste en obtener los recursos humanos requeridos para el completamiento del proyecto.

*Desarrollo del equipo de proyecto:* Consiste en mejorar las competencias y la interacción de los miembros del equipo para mejorar el rendimiento del proyecto.

*Gestión del equipo de proyecto:* Consiste en seguir el rendimiento de los miembros del equipo, proveyendo retroalimentación, resolviendo problemas y coordinando los cambios para mejorar el rendimiento del proyecto.

Algunas interacciones pueden resultar en la necesidad de la planificación adicional, por ejemplo:

- Luego de que los miembros iniciales del equipo elaboren el WBS, puede necesitarse la adquisición de más miembros para el equipo de proyecto.
- Mientras son adquiridos miembros adicionales para el equipo de proyecto, el nivel de experiencia de estos podría incrementar o disminuir algún riesgo del proyecto, creando la necesidad de planificación adicional de los riesgos.
- Cuando la duración de las actividades es estimada antes de que se conozcan todos los miembros del equipo de proyecto, los niveles de competencia reales de los miembros que sean adquiridos pueden causar que la duración de alguna actividad o el Schedule cambien. **(4)**

### **1.3.2.7 Gestión de las Comunicaciones**

La gestión de las comunicaciones del proyecto es el área de conocimiento que emplea los procesos requeridos para asegurar la generación, recolección, distribución, almacenamiento y finalmente disposición apropiada y en tiempo de la información del proyecto. Los procesos de la gestión de las comunicaciones del proyecto proveen vínculos críticos entre las personas y la información que son necesarios para que las comunicaciones sean exitosas. Los administradores de proyectos pueden gastar una increíble cantidad de tiempo comunicándose con el equipo del proyecto, con los stakeholders, con los clientes y los patrocinadores. Todo el que este involucrado en el proyecto debe entender como las comunicaciones pueden tener un efecto en el proyecto como un todo. Los procesos de la gestión de las comunicaciones del proyecto son:

*Planeamiento de las comunicaciones:* Consiste en determinar la información y las comunicaciones que necesitan los stakeholders del proyecto.

*Distribución de la información:* Consiste en mantener disponible la información que necesitan los stakeholders, en lo que respecta al tiempo.

*Reporte de rendimiento:* Consiste en recolectar y distribuir la información acerca del rendimiento. Esto incluye el reporte de estado, las medidas de progreso y los pronósticos.

*Gestionar los stakeholders:* Consiste en gestionar las comunicaciones para satisfacer los requisitos y solucionar las diferencias con los stakeholders del proyecto. **(4)**

### **1.3.2.8 Gestión del Aprovisionamiento**

La gestión del aprovisionamiento del proyecto incluye los procesos de compra o adquisición de los productos, servicios o resultados necesitados de fuentes externas al proyecto para ejecutar algún trabajo.

La gestión del aprovisionamiento del proyecto incluye la gestión de los contratos y el control de los cambios en los procesos requeridos para administrar los contratos o las órdenes de compra efectuadas por los miembros del equipo de proyecto que están autorizados para hacerlo.

Los procesos de la gestión del aprovisionamiento son:

*Planificación de las compras y adquisiciones:* Consiste en determinar que debe ser comprado o adquirido además de determinar como y cuando.

*Planificación de los contratos:* Consiste en documentar los requerimientos de productos, servicios o resultados que se necesitan e identificar proveedores potenciales.

*Solicitud de respuesta a los proveedores:* Consiste en obtener información, cotizaciones, posturas, ofertas o proposiciones de los posibles proveedores.

*Selección de los proveedores:* Consiste en revisar las ofertas, seleccionando entre los proveedores potenciales los que sean mas convenientes al proyecto, y negociar un contrato escrito con cada proveedor.

*Administración de contratos:* Consiste en gestionar los contratos y las relaciones con los proveedores, revisando y documentando el comportamiento de los proveedores, y gestionar los cambios referentes los contratos.

*Cierre de contratos:* Consiste en cerrar cada contrato aplicable al proyecto o a alguna fase del proyecto.

La gestión del aprovisionamiento del proyecto involucra los contratos, que son documentos legales que vinculan al comprador con el proveedor. Un contrato es un acuerdo mutuo vinculante que obliga al proveedor a cumplir con la entrega de los productos, servicios o resultados especificados y obliga al comprador a cumplir con la entrega de un componente monetario u otro componente valuable. Un contrato es una relación legal sujeta a ser resuelta en una corte de justicia.

Las distintas actividades involucradas en la gestión del aprovisionamiento del proyecto forman el ciclo de vida de cada contrato. Gestionando activamente el ciclo de vida de cada contrato y redactando cuidadosamente los términos y condiciones de estos es una manera efectiva de evitar o mitigar algunos riesgos identificados del proyecto. **(4)**



Un proyecto complejo puede implicar la gestión de muchos contratos o subcontratos simultáneos o secuenciados. En esos casos, el ciclo de vida de cada contrato podría terminar durante cualquier fase del ciclo de vida del proyecto.

### **1.3.3 Situación Problemática**

La institución líder a nivel mundial en la Gestión de Proyectos como disciplina, el Project Management Institute, ha generado un amplio conocimiento de cómo combinar las habilidades, técnicas y herramientas para lograr un resultado exitoso al terminar el ciclo de vida de un proyecto, si tenemos en cuenta que la cantidad de variables a tener en cuenta cuando se está gestionando un proyecto es considerablemente alta, resulta sencillo concluir que gestionar un proyecto es una tarea bastante difícil, de hecho, sin el apoyo de las TICs, gestionar un proyecto implica un costo, en tiempo y recursos, mucho mayor que cuando se involucran las tecnologías de la información y las comunicaciones, sin embargo, la mayoría de las herramientas que se encuentran en el mercado son propietarias y su implantación en cualquier organización resulta verdaderamente costosa, en principal medida por los pagos, por concepto de licencias, que son necesarios efectuar a las compañías desarrolladoras de estas herramientas. Sería muy conveniente el uso de herramientas con licenciamiento GPL para, de alguna manera, amortizar los costos por licenciamiento, sin embargo la mayoría de las herramientas presentes en el mercado y específicamente las desarrolladas en software libre no abarcan de manera explícita un grupo de procesos de la Gestión de Proyectos, como lo son los incluidos en la Gestión de la Calidad, del Aprovechamiento y los Riesgos.

## **1.4 Estado del Arte**

### **1.4.1 Herramientas de Apoyo a la Gestión**

#### **1.4.1.1 Microsoft Project**

Esta herramienta basada en Windows es comúnmente utilizada para realizar estimaciones de duración desde proyectos de envergadura hasta simples planes de estudios. Permite realizar variaciones a las estimaciones debido a los múltiples imprevistos que pueden presentarse en una obra a largo plazo, e incluso en algunos casos puede sugerir soluciones a las demoras. Para su uso es imprescindible la descomposición del proyecto en tareas, estas a su vez en subtareas y así sucesivamente hasta que sea posible. Tiene en cuenta personas (numero y función de cada una), recursos materiales, tareas a realizar, dependencias entre tareas, fechas de entrega.

Como software propietario que es, una de las mayores desventajas que tiene es el costo de su utilización ya sea por los pagos de las licencias pertenecientes a Microsoft o los cursos que deben recibir el personal que debe usar la herramienta, además, MsProject es una aplicación escritorio que no permite un trabajo en equipo fluido, principalmente si el equipo de trabajo se encuentra disperso físicamente hablando, lo que complica que todo el equipo tenga la información del proyecto disponible y actualizada.

#### **1.4.1.2 Idinet**

Es una herramienta desarrollada con la plataforma Microsoft Windows Sharepoint Services, para trabajar en entorno Web. Idinet permite la gestión de proyectos, posibilitando la vinculación de los procesos que se llevan a cabo en momentos determinados dentro del desarrollo del proyecto. Gestiona los permisos sobre el acceso a la información para que los cambios puedan realizarse solamente por el personal autorizado. Permite llevar un registro histórico de los cambios realizados en las actividades del proyecto.

Posibilita la gestión documental diferenciada para cada uno de los proyectos, de forma que todos los documentos relacionados con una determinada actividad siempre estarán perfectamente localizados y clasificados.

Es un software propietario, desarrollado sobre Windows, cuya ejecución requiere de la instalación de Windows 2003 Server y SQL Server, con las obligaciones financieras que, en cuanto a licencias, esto implica.

#### **1.4.1.3 GanttPV**

Es un programa gratuito, que cuenta con una apariencia sencilla, que actualmente se encuentra disponible para Windows, MacOS y Linux.

GanttPV permite definir tareas, sus duraciones, dependencias, fechas de inicio de estas, e incluso días en los que no se trabajará. Usando esta información GanttPV permite calcular la fecha de culminación de las tareas definidas y permite la creación de un diagrama de Gantt que ayuda al entendimiento del estado del proyecto.

GanttPV ayuda a la identificación de los recursos necesarios por una tarea determinada así como a la asignación de un recurso a una tarea en particular. Esta herramienta permite, además, priorizar las tareas que componen al proyecto.

De manera general GanttPV es una herramienta que implementa algunos procesos contenidos en las áreas de conocimientos Gestión del Tiempo y Gestión del Costo principalmente, además tiene a su favor el ser una herramienta open source y gratuita.

#### **1.4.1.4 dotProject**

DotProject es una herramienta Web de Gestión de Proyectos, escrita en PHP con la base de datos en MySQL. Tiene una plantilla y diseño excelente.

Se separa de las demás herramientas ofreciendo un punto de vista ignorado por varias herramientas de este tipo: en vez de gestionar proyectos, con dotProject, se pueden gestionar varias empresas, todas involucradas en varios proyectos.

Las unidades fundamentales de esta herramienta son: empresas, proyectos, actividades y usuarios. Los proyectos, tareas y usuarios están vinculados a una compañía. La gestión de actividades es el componente crítico de dotProject, además permite ver las actividades de una manera gráfica, ordenadas jerárquicamente con sus dependencias y solapamientos, permite adjuntar archivos a las actividades y a los proyectos brindando un versionado básico de los mismos, permite la creación de foros de discusión dentro de los proyectos, así como la asignación de recursos a las distintas actividades.

### **1.4.2 ERPs**

#### **1.4.2.1 SAP ERP**

SAP ERP es una aplicación integrada para la planificación de recursos empresariales, producida por la más grande corporación de software en Europa y la tercera en el mundo SAP AG.

SAP ERP cuenta con un sinnúmero de funcionalidades, que pueden ser de mucha ayuda a la hora de analizar un negocio, optimizar el manejo de las finanzas, gestionar los recursos humanos etc.

SAP ERP incluye cuatro soluciones individuales que se combinan a fin de proporcionar una base sólida para la gestión de los procesos del negocio:

- SAP ERP Financials, para controlar la economía del proyecto, como punto estratégico en la realización del mismo.
- SAP ERP Human Capital Management, para gestionar los recursos humanos con el fin de maximizar el potencial de los trabajadores.
- SAP ERP Operations, para el control de los recursos con el fin de mantener el presupuesto en el rango definido.
- SAP ERP Corporate Services, para la gestión de los procesos del negocio.

De manera general, SAP ERP, es un sistema que puede satisfacer la mayoría de las necesidades operativas en el desarrollo de un proyecto específico. Las desventajas de esta aplicación consisten básicamente en lo cara que puede resultar la inversión económica que habría que hacer para la implantación del paquete, así como el esfuerzo en tiempo y dinero para capacitar al personal para el uso de la herramienta.

#### **1.4.2.2 ERP Primavera**

El ERP de Primavera BSS se caracteriza por su modularidad, un alto nivel de flexibilidad en la ejecución de procesos y tareas mediante alertas e indicadores de negocio, junto con elevados niveles de integración entre los módulos. A esto se une la disposición de capacidades que agilizan el acceso y la explotación de los datos y la navegación entre ventanas relacionadas.

Las soluciones que conforman a ERP Primavera están basadas en tecnologías estándar, que permiten crear una plataforma potente y estable, que puede ser ampliada y enriquecida con otras aplicaciones y desarrollos, también basados en estándar. Todos los productos Primavera están desarrollados siguiendo el Standard "Windows DNA - Windows Distributed interNet Applications Architecture". Se trata de una arquitectura de 3 capas (n-capas), orientada a los servicios, donde están perfectamente identificados y separados los procesos de presentación, lógica de las aplicaciones y acceso a datos.

En su creación se empleó el la plataforma .NET de Microsoft, y almacenamiento sobre base de datos SQL Server. El sistema admite la creación de nuevos módulos y funcionalidades.

Al igual que SAP, Primavera es un software propietario, que se ejecuta, sobre Windows, con la implicación financiera que se deduce de esto, además es bastante complicada de aprender aunque ya después es de muy fácil uso.

#### **1.4.2.3 PeopleSoft**

Luego de su adquisición por Oracle, PeopleSoft afianzó su 2do lugar en el mercado mundial de los ERPs, solo superado por SAP, su arma más fuerte son los módulos de gestión de los recursos humanos, aunque se han dado pasos serios en la orientación de la herramienta al área de los servicios, implementando funcionales referentes al control del costo y los recursos.

Como ventaja tiene el no estar atado a un gestor de base de datos en particular, pues puede funcionar con bases de datos montadas en Oracle, Microsoft SQL Server, Informix, Sybase, IBM DB2 entre otros.

En su contra tiene el costo de adquisición e implementación, además de que eventualmente podría necesitar el apoyo de algún servicio externo pues como ya fue dicho, su principal profundidad se encuentra en la Gestión de los RRHH y no así en otras áreas.

#### **1.4.2.4 Microsoft Dynamics NAV**

Este ERP ha pasado por un largo proceso de evolución "Navision Financials", "Navision Attain", "Microsoft Business Solutions Navision" y el actual (2006) "Microsoft Dynamics NAV" hacen referencia al mismo producto.

Diseñado para pequeñas y medianas empresas, Microsoft Dynamics NAV es una solución de gestión empresarial integrada. Permite mejorar los puntos fuertes del negocio, sacar el máximo partido al tiempo y recursos, tener acceso a la información en cualquier idioma, ayudando a su vez a optimizar el negocio y aumentar la productividad. La implementación es rápida y eficaz, resulta sencillo personalizarla, mantenerla y compatibilizarla con otros sistemas. Sus fortalezas clave se apoyan en la gestión financiera, fabricación, distribución, gestión de relaciones, de servicios, comercio electrónico y análisis.

Como valor agregado Microsoft Dynamics NAV, permite el acceso a su código fuente, por lo que lo hace un producto altamente personalizable. El lenguaje utilizado en su programación es

el C/AL, un lenguaje similar al Pascal, independientemente de que sea open source, este producto no es gratuito, y el costo para su implementación en una empresa es la principal desventaja que tiene.

#### **1.4.2.5 OpenPro**

OpenPro es un software ERP desarrollado en tecnología "Open Source". OpenPro está escrito en php, con una arquitectura abierta y corre sobre cualquier hardware y cualquier gestor de bases de datos SQL.

OpenPro permite gestionar muchos de las operaciones claves de un negocio en un solo sistema integrado, que incluye, gestión de las relaciones con los clientes, cadena de aprovisionamiento, gestión y completamiento de las órdenes, gestión de inventario, finanzas, Libro Mayor general, cuentas por pagar, créditos, comercio electrónico y gestión del sitio Web de la empresa, fabricación, activos fijos, gestión de las garantías, y productividad de los empleados.

Estas funcionalidades están agrupadas en módulos o subaplicaciones que son: Finanzas, Fabricación, CRM (gestión de las relaciones con los clientes), Comercio Electrónico, Distribución.

### **1.5 Conclusiones**

En este capítulo se profundizó en los conceptos más importantes referentes a la Gestión de Proyectos. Se describieron las distintas áreas de conocimientos que componen el grupo de "buenas prácticas" de la Gestión de Proyectos, y se hizo un estudio breve de algunas herramientas que informatizan un grupo de procesos de la Gestión de Proyectos.

Luego de haber estudiado los conceptos y definiciones más importantes, así como las condiciones actuales en las que se enmarca el objeto de estudio, se fundamenta la conclusión de proponer una herramienta en Software Libre, que informatice un amplio espectro de procesos de la Gestión de Proyectos con el objetivo de optimizar el proceso de toma de decisiones.

## **CAPÍTULO 2: Tendencias y tecnologías actuales a desarrollar.**

### **2.1 Introducción**

A medida que avanza el tiempo el desarrollo de las TICs se hace cada vez más vertiginoso, cada día surgen nuevas herramientas de desarrollo y se perfeccionan las ya existentes, generando un amplio espectro de posibles metodologías, lenguajes y herramientas de desarrollo que se pueden usar para la creación de aplicaciones basadas en las TICs.

El siguiente capítulo realiza una enumeración de las características de los medios que se utilizarán para la documentación y el desarrollo del sistema que se propone en el trabajo.

### **2.2 Metodologías de Desarrollo de Software**

#### **2.2.1 Microsoft Solution Framework (MSF)**

Microsoft Solution Framework (MSF) es un enfoque disciplinado para proyectos basados en tecnología, define un grupo de principios, modelos, disciplinas, conceptos, guías y practicas que pretenden facilitar la consecución de los objetivos durante el ciclo de vida del desarrollo del software.

MSF provee una metodología flexible y escalada que puede ser adaptada para cumplir con las necesidades de cualquier proyecto (sin importar tamaño o complejidad) al planificar, construir y desplegar soluciones tecnológicas. La filosofía de MSF sostiene que no existe ninguna estructura o proceso que coincida de manera óptima con los requisitos de todos los proyectos. Como metodología, MSF provee una guía sin imponer detalles que podrían estar limitados a un estrecho rango de posibles escenarios. Los componentes de MSF pueden ser aplicados de manera individual o colectiva para aumentar la probabilidad de éxito en los siguientes tipos de proyecto:

- Proyectos de desarrollo de software, incluyendo aplicaciones Web y de e-commerce (comercio electrónico).
- Proyectos de despliegue de infraestructuras, incluyendo aplicaciones de escritorio, actualizaciones de sistemas operativos, mensajería corporativa y sistemas de configuración y administración.

- Proyectos de aplicaciones integradas, incluyendo suites de productividad personal, planificación de recursos corporativos (ERP por sus siglas en inglés) y soluciones de gestión de proyectos.
- Cualquier combinación (por compleja que sea) de las anteriores.

La guía que MSF ofrece para estos distintos tipos de proyectos se enfoca en administrar a las “personas y procesos” y a los elementos tecnológicos que incluyen la mayoría de los proyectos.

Como se dice anteriormente MSF como metodología contiene múltiples componentes que pueden ser utilizados individualmente o como un todo integrado. Cuando se usan estos componentes de manera colectiva crean un enfoque sólido pero flexible para la ejecución exitosa de proyectos tecnológicos. La siguiente lista define a estos componentes.

- **Principios Fundacionales de MSF.** Es el núcleo de principios en los que se basa la metodología MSF. Estos expresan valores y estándares que son comunes para todos los elementos de MSF.
- **Modelos de MSF.** Son descripciones esquemáticas o “mapas mentales” de la organización de los equipos del proyecto y sus procesos.
- **Disciplinas de MSF.** Son las áreas de práctica que usan un grupo específico de métodos, términos y enfoques.
- **Conceptos claves de MSF.** Son las ideas que soportan a los principios y las disciplinas de MSF y que son mostradas a través de prácticas específicas.
- **Prácticas de MSF.** Son las prácticas que han sido probadas efectivamente en proyectos tecnológicos bajo una variedad de condiciones reales.



- **Recomendaciones de MSF.** Son prácticas y guías opcionales pero recomendadas a la hora de aplicar los modelos y las disciplinas.

## 2.2.2 Extreme Programming (XP)

Extreme Programming (XP) es un enfoque disciplinado para el desarrollo de software. En sus aproximadamente 8 años de edad ya ha sido probado en varias compañías e industrias de diversos tamaños en todo el mundo.

Esta metodología esta diseñada para facilitar la entrega del software requerido por el cliente cuando el cliente lo requiere. XP motiva a los desarrolladores a responder a los cambiantes requerimientos del cliente, incluso cuando el ciclo de vida del desarrollo se encuentre en su etapa final.

XP además enfatiza en el trabajo en equipo. Administradores, clientes y desarrolladores son parte de un equipo que tiene como objetivo la entrega de un software con calidad. El enfoque de XP para el desarrollo de software tiene 3 características esenciales:

- **Comunicación.** Los desarrolladores que concibe XP se comunican activamente con sus clientes y con sus compañeros desarrolladores.
- **Simplicidad.** Los desarrolladores que concibe XP mantienen el diseño del software tan simple y limpio como sea posible.
- **Retroalimentación.** Los desarrolladores que concibe XP se retroalimentan realizándole pruebas al software desde el primer día de desarrollo. Estos entregan el producto a los clientes tan pronto como es posible e implementan los cambios requeridos por los clientes.

La filosofía de XP de una comunicación activa con los clientes para retroalimentar el proceso de desarrollo, así como el obtener en breve tiempo un producto que tenga alguna funcionalidad, para después ir agregándole otras hasta que se cumpla con los requisitos del cliente, ha permitido que XP gane muchos adeptos a nivel mundial pues es más fácil confiar en una metodología que produce algo que se puede usar e instalar de manera paralela al proceso de desarrollo.

Las técnicas de XP se dividen en 4 ámbitos:

### **1- Planificación**

- Historias de Usuarios.
- Sacar nuevas versiones con frecuencia.
- Iteraciones.
- Trabajo en Equipo.

### **2- Diseño**

- Regla KISS.
- Nomenclatura.
- Seguir estrictamente el plan.
- Refactorización.

### **3- Codificación**

- Comunicación con el usuario.
- Estándares de codificación.
- Programación en pareja.
- Pruebas de unidad.
- Integración continua.
- Propiedad colectiva del código.
- Solo optimizar cuando es necesario.
- Cuidado con hacer muchas horas.

### **4- Pruebas**

- Testing en cada iteración.

### **2.2.3 Rational Unified Process (RUP)**

Rational Unified Process (RUP) es una metodología de desarrollo de software que puede ser utilizada para un amplio espectro de tipos de proyectos de software, en distintas áreas de aplicación, en distintos tipos de organizaciones, con diferentes niveles de competencia y en proyectos de distintos tamaños.

RUP provee un enfoque disciplinado en cuanto a la asignación de tareas y responsabilidades dentro del equipo de desarrollo del software. El objetivo es lograr un producto de alta calidad y que cumpla con las necesidades del usuario, en tiempo y dentro del presupuesto aprobado.

RUP se puede mirar de 2 formas:

- **Horizontalmente.** Representa el tiempo y muestra los aspectos del ciclo de vida del proceso a lo largo de su desarrollo.
- **Verticalmente.** Representa las disciplinas, bajo las cuales se agrupan actividades de una manera lógica de acuerdo a su naturaleza.

La primera forma muestra el aspecto dinámico del proceso conforme se va desarrollando, se expresa en fases, iteraciones e hitos. La segunda forma representa lo estático del proceso, cómo es descrito en términos de componentes del proceso, disciplinas, actividades, flujos de trabajo, artefactos y roles.

Los aspectos distintivos de RUP están encapsulados en 3 conceptos claves que a su vez son los que hacen único a RUP:

### **1- Dirigido por casos de uso:**

Un sistema de software es creado siempre para servir a usuarios. Por lo tanto es necesario definir bien qué es lo que quieren y necesitan los futuros usuarios. Un caso de uso sería, entonces, una pieza de funcionalidad del sistema que devuelve un resultado de valor para el usuario. Los casos de usos cubren los requerimientos funcionales. El conjunto de casos de usos constituyen el modelo de casos de uso que describe la funcionalidad completa del sistema.

Sin embargo, aunque el proceso este dirigido por los casos de uso, estos no son elegidos de manera aislada. Los casos de uso son desarrollados a la par con la arquitectura del sistema.

### **2- Centrado en la arquitectura.**

El concepto de arquitectura de software involucra los aspectos estáticos y dinámicos más significativos del sistema. La arquitectura surge de las necesidades de la empresa, tal como están reflejadas en los casos de uso. Pero la arquitectura además se ve influenciada por otros factores como la plataforma en la que el software debe ejecutarse, la cantidad de componentes reutilizables disponibles y los requerimientos no funcionales de manera general. La arquitectura es la vista del diseño completo con las características más importantes, dejando los detalles de lado.

Cada producto tiene función y forma. Uno solo de los dos no es suficiente. Por una parte, los casos de uso deben, cuando son realizados, acomodarse en la arquitectura. Por otro lado, la arquitectura debe dejar espacio para la realización de todos los casos de usos en el presente y en el futuro. La arquitectura y los casos de usos deben evolucionar en paralelo.

### **3- Iterativo e Incremental.**

Desarrollar un software con un objetivo comercial es algo bien complejo que puede demorar hasta varios años. Es lógico dividir el trabajo en pedazos más pequeños. Cada pedazo es una iteración que finaliza en un incremento. Las iteraciones se refieren a pasos en el flujo de trabajo, los incrementos se refieren al crecimiento del producto. Para ser más efectivo las iteraciones deben estar controladas, deben ser seleccionadas y llevadas a cabo siguiendo un plan.

Los desarrolladores basan la elección de que van a implementar en una iteración en 2 factores. Primero, la iteración trata con un grupo de casos de uso que, en su conjunto, extienden la usabilidad del producto. Segundo, la iteración trata con los riesgos más importantes. Las iteraciones sucesivas construyen los artefactos a partir del estado en el que fueron dejados en la iteración anterior.

En cada iteración, los desarrolladores identifican y especifican los casos de uso relevantes, crean el diseño usando la arquitectura como guía, implementan el diseño en componentes y verifican que los componentes satisfacen los casos de uso. Si una iteración cumple sus objetivos el desarrollo continúa con la próxima iteración.

## **2.2.4 Fundamentación de la metodología seleccionada (RUP)**

El transcurso del tiempo ha decretado una realidad, para cumplir con los requerimientos del cliente a la hora de entregar un producto de software, es necesario el uso de una metodología de desarrollo que nos permita darle una organización al trabajo, para así realizar la entrega con las variables de tiempo, costo y calidad dentro de los valores previamente acordados.

Para elegir la metodología de desarrollo adecuada es necesario mirar al proyecto que se quiere desarrollar desde adentro para determinar su alcance, posible duración y complejidad. Luego de analizar lo que se propone con el trabajo y teniendo en cuenta el posible volumen de procesos a informatizar se llega a la conclusión de que la metodología más factible sería RUP.

RUP es especialmente efectivo para los proyectos de gran magnitud como este, además de poseer una gran adaptabilidad cuando se habla de proyectos a largo plazo, por otro lado la filosofía de casos de usos y arquitectura es perfecta para describir la amplia gama de funcionalidades que requiere implementar un proyecto como este. Por último, RUP da la posibilidad de generar la documentación necesaria que garantizaría la continuidad del proyecto.

## **2.3 Lenguaje de Modelación: Unified Modeling Language (UML)**

Las grandes aplicaciones empresariales, aquellas que informatizan los procesos empresariales más importantes y que mantienen a la empresa funcionando deben ser más que en puñado de módulos llenos de código. Deben estar estructuradas de una manera que les permitan tener escalabilidad, seguridad y ejecución robusta bajo condiciones de “estrés”, y su arquitectura debe ser definida con la suficiente claridad para que los programadores de mantenimiento puedan, rápidamente, encontrar y arreglar los “bugs” que muestre la aplicación mucho después que los programadores originales entregasen el producto. Se hace mención a las grandes aplicaciones empresariales en primer lugar porque, además de ser la propuesta de este trabajo, estructurar bien el trabajo es una manera de manejar la complejidad de proyectos complejos.

Modelar es una parte esencial de los proyectos grandes de software y un medio sumamente útil para proyectos de tamaño medio e incluso pequeños. Un modelo juega un rol análogo en el desarrollo de un software que el que jugarían los planos en la construcción de un edificio “rasca cielos”. Mediante el uso de modelos, los responsables del desarrollo de un software

determinado pueden asegurarse de que las funcionalidades que ofrecerá el sistema están completas y son las correctas, que se cumplen las necesidades del usuario y que los requisitos de robustez, seguridad etc. son cumplidos antes de que la implementación del sistema avance y haga mas difícil y costoso corregir los errores.

El lenguaje unificado de modelación (UML por sus siglas en inglés) es un lenguaje estándar para especificar, visualizar, construir y documentar los artefactos de los sistemas de software, aunque es útil además para modelar negocios y otros sistemas que no son de software. UML contiene una colección de las mejores prácticas ingenieriles que han sido probadas exitosamente en el modelado de sistemas grandes y complejos. UML es una parte importante en el desarrollo de software orientado a objetos y en el desarrollo de procesos del software. UML generalmente usa notaciones gráficas para expresar el diseño de proyectos de software. El uso de UML ayuda a los equipos de proyecto a comunicarse, explorar diseños potenciales y validar el diseño arquitectónico del software.

Los Objetivos Primarios de UML son:

- Proveer a los usuarios con un lenguaje listo para usarse, bien expresivo y visual, para facilitar el desarrollo de modelos significativos.
- Ser independiente de cualquier lenguaje de programación y proceso de desarrollo.
- Proveer una base formar para el entendimiento del lenguaje de modelación.
- Soportar conceptos de desarrollo de alto nivel como colaboraciones, plataformas, patrones y componentes.
- Integrar las mejores prácticas ingenieriles del software.

El aumento del valor estratégico del software para muchas empresas en el mundo ha provocado la búsqueda de la forma óptima en la producción de software, así como mejorar la calidad de estos y reducir el costo y el tiempo para la salida del software al mercado. Las empresas buscan además una manera para manejar la complejidad de algunos sistemas al tiempo que estos aumentan en alcance y escala. De manera particular las empresas reconocen la necesidad de resolver los problemas arquitectónicos recurrentes, como la distribución física, la concurrencia, la replicación y la seguridad. Adicionalmente los desarrolladores Web tratando de hacer las cosas de una manera simple han experimentado grandes problemas arquitectónicos. El Lenguaje Unificado de Modelación (UML) fue diseñado para responder a estas necesidades.

## **2.4 Arquitectura de Software**

### **2.4.1 Modelo Vista Controlador**

El Modelo Vista Controlador (MVC) es un patrón de arquitectura usado en la Ingeniería de Software. El uso correcto de este patrón aísla la lógica del negocio de las interfaces de usuario, esto permite modificar la apariencia visual de la aplicación o las reglas subyacentes del negocio, sin afectar ninguna de las dos. En MVC, el Modelo representa la información, o los datos de la aplicación, así como las reglas del negocio que se usan para manipular estos datos, la Vista corresponde a los elementos de la interfaz de usuario como el texto y los componentes usados para comunicarse con él, por último, el Controlados maneja la comunicación entre las acciones ejecutadas en la Vista y las repercusiones que deben tener en el Modelo.

Es común dividir una aplicación en varias capas: presentación (interfaz de usuario), lógica del negocio y acceso a datos. En MVC la capa de presentación también se divide en vista y controlador. MVC abarca más en la arquitectura de una aplicación de lo que normalmente lo hacen los patrones de diseño.

Modelo:

Consiste en la representación del dominio específico de la información donde la aplicación opera. Muchas aplicaciones usan mecanismos persistentes de almacenamiento, bases de datos por ejemplo, para guardar la información. MVC no menciona específicamente a la capa de acceso a datos porque está comprendida en el Modelo.

Vista:

Transforma al modelo en una forma entendible para la interacción con el usuario. Múltiples vistas con varios propósitos pueden existir para un mismo modelo.

Controlador:

Procesa y responde ante los eventos, normalmente acciones del usuario, y puede invocar cambios en el Modelo.

MVC se puede ver con frecuencia en aplicaciones Web, donde la vista es la página html y el controlador es el código que provee los datos dinámicos y genera el contenido dentro de la

página html. Finalmente, el modelo es representado por el contenido real, usualmente almacenado en una base de datos o un XML y las reglas del negocio que transforman esos datos en contenido basándose en las acciones del usuario.

## **2.5 Lenguajes de Desarrollo Web**

### **2.5.1 ASP**

Active Server Pages o como comúnmente se le conoce: ASP es una tecnología desarrollada por Microsoft que permite desarrollar páginas Web interactivas y dinámicas. ASP usa el código del lado del servidor para producir páginas Web dinámicamente. El lenguaje de codificación por defecto de ASP es VBScript, pero se pueden usar otros lenguajes de codificación como el Jscript (Versión de JavaScript desarrollada por Microsoft).

Las páginas ASP tienen la extensión .asp en lugar de .htm, cuando una página ASP es solicitada el servidor Web la interpreta y produce una página HTML que es enviada al cliente. Las páginas ASP se deben encontrar en un servidor Web que soporte ASP, como ASP fue creado por Microsoft estas páginas corren mejor en un servidor Internet Information Server (IIS) que es el servidor Web desarrollado por Microsoft.

### **2.5.2 ASP.NET**

ASP.NET no es la siguiente versión de ASP, es toda una nueva filosofía de desarrollo Web. ASP.NET permite usar lenguajes de programación muy completos como C# o VB.NET para construir aplicaciones Web de una manera sencilla.

ASP.NET procesa todo el código en el servidor, cuando el código ASP.NET ha sido procesado el servidor retorna la página HTML resultante al cliente. Incluso aunque la necesidad del uso de código HTML es una limitante, ASP.NET es una buena opción para el uso del paradigma de la programación orientada a objetos en Internet.

Una característica magnífica de ASP.NET son los Web Services. Con el uso de esos se puede, literalmente hablando, tener varios pedazos de la aplicación en diferentes servidores en todo el mundo y que la aplicación funciones exactamente igual.



ASP.NET incluye una amplia librería de clases que ha sido desarrollada por Microsoft. Esta librería encapsula un gran cantidad de funciones que hacen más fácil el trabajo de los desarrolladores.

Algunas Ventajas de ASP.NET:

- Reduce drásticamente la cantidad de código a escribir para desarrollar aplicaciones grandes
- Hace del desarrollo un proceso más sencillo mediante un modelo de programación dirigido por eventos y del lado del servidor.
- Hace que las páginas sean más sencillas de escribir y mantener dado que el código fuente y el HTML están juntos.
- El código se ejecuta siempre en el servidor lo que hace que le da poder y flexibilidad a las páginas.
- El código fuente se compila la primera vez que la página es solicitada lo que hace que la ejecución sea más rápida ya que el servidor guarda la versión compilada para la próxima vez que la página sea solicitada.
- El servidor Web monitorea continuamente las páginas, los componentes y las aplicaciones que corren en el, si detecta fallos de memoria, ciclos infinitos u otra actividad "ilegal" del software la interrumpe y se reinicia automáticamente.
- Facilita la validación de la información entrada por los usuarios sin la necesidad de escribir código.
- Funciona más rápido con volúmenes grandes de usuarios conectados sin enfrentar problemas de funcionamiento.

### **2.5.3 PERL**

Es un lenguaje de programación muy utilizado para construir aplicaciones CGI para la Web. Perl es un acrónimo de Practical Extracting and Reporting Language, que viene a indicar que se trata de un lenguaje de programación muy práctico para extraer información de archivos de texto y generar informes a partir del contenido de los ficheros.

Es un lenguaje libre de uso, eso quiere decir que es gratuito. Antes estaba muy asociado a la plataforma Unix, pero en la actualidad está disponible en otros sistemas operativos como Windows.

Perl es un lenguaje de programación interpretado, al igual que muchos otros lenguajes de Internet como Javascript o ASP. Esto quiere decir que el código en Perl no se compila, sino que cada vez que se quiere ejecutar se lee el código y se pone en marcha interpretando lo que hay escrito. Además es extensible a partir de otros lenguajes, ya que desde Perl podremos hacer llamadas a subprogramas escritos en otros lenguajes. También desde otros lenguajes podremos ejecutar código Perl.

Perl está inspirado a partir de lenguajes como C, sh, awk y sed (algunos provenientes de los sistemas Unix), pero está enfocado a ser más práctico y fácil que estos últimos. Es por ello que un programador que haya trabajado con el lenguaje C y los otros tendrá menos problemas en entenderlo y utilizarlo rápidamente. Una diferencia fundamental de Perl con respecto a los otros lenguajes es que no limita el tamaño de los datos con los que trabaja, el límite lo pone la memoria que en ese momento se encuentre disponible.

No establece ninguna filosofía de programación concreta. No se puede decir que sea orientado a objetos, modular o estructurado aunque soporta directamente todos estos paradigmas; su punto fuerte son las labores de procesamiento de textos y archivos.

#### **2.5.4 JSP**

JSP (Java Server Pages) es una tecnología del lado del servidor, es una extensión de la tecnología de servlets de Java que fue desarrollada por SUN. JSP tiene la capacidad de codificación dinámica que trabaja junto al código HTML, separando la parte lógica de la página de los elementos estáticos (el diseño de la página). Incrustado en la página HTML, el código fuente de Java y sus extensiones ayudan a hacer al HTML más funcional. JSP no está restringido a una plataforma específica del servidor.

El motor de las páginas JSP está basado en los servlets de Java -programas en Java destinados a ejecutarse en el servidor-, aunque el número de desarrolladores que pueden afrontar la programación de JSP es mucho mayor, dado que resulta mucho más sencillo aprender que los servlets.

En JSP se crean páginas de manera parecida a como se crean en ASP o PHP. Se generan archivos con extensión .jsp que incluyen, dentro de la estructura de etiquetas HTML, las sentencias Java a ejecutar en el servidor. Antes de que sean funcionales los archivos, el

motor JSP lleva a cabo una fase de traducción de esa página en un servlet, implementado en un archivo class. Esta fase de traducción se lleva a cabo habitualmente cuando se recibe la primera solicitud de la página .jsp, aunque existe la opción de precompilar el código para evitar ese tiempo de espera la primera vez que un cliente solicita la página.

### **2.5.5 PHP**

PHP (Personal Home Page) es conocido oficialmente como “PHP: HyperText Preprocessor”. Es un lenguaje de programación del lado del servidor que es escrito usualmente en un contexto HTML. A diferencia de una página HTML ordinaria, el código PHP no es enviado al cliente por el servidor, el código es ejecutado por el módulo PHP que se encuentra instalado en el servidor, el código HTML no se toca pero el código PHP es interpretado y ejecutado por dicho módulo para crear el documento HTML final que será enviado al cliente. Usando PHP se puede realizar queries a bases de datos, crear imágenes, leer y escribir en archivos y comunicarse con servidores remotos.

Apache es actualmente el servidor Web más usado mundialmente y PHP puede ser compilado como un módulo de Apache, lo que ha hecho que PHP gane muchos adeptos a nivel mundial.

Esto no significa que PHP no puede trabajar en otro ambiente. De hecho PHP es compatible con una amplia lista de servidores Web y gestores de bases de datos. El alza en la popularidad de PHP ha coincidido con un cambio de enfoque en la programación Web. Mientras a mediados de los noventa estaba bien construir sitios, incluso sitios relativamente grandes, con cantidad de páginas HTML individuales, ahora la tendencia es hacia el uso del poder de las bases de datos para gestionar el contenido de estas páginas de una manera más efectiva.

### **2.5.6 Fundamentación del lenguaje seleccionado (PHP)**

Como producto de código abierto, PHP está soportado por un talentoso equipo de producción y una comunidad de usuarios comprometida. Además, PHP puede correr en todos los sistemas operativos importantes.

La velocidad de desarrollo es algo importante también. PHP permite separar el código HTML de los elementos codificados en PHP lo que disminuirá el tiempo de desarrollo de varias

aplicaciones. Esto no solo le facilita el trabajo a los desarrolladores sino que también elimina obstáculos cuando se trata de conjugar la lógica con un diseño efectivo y flexible.

Luego de comparar las principales ventajas de PHP con las características de los lenguajes de programación más populares a nivel mundial, se escoge PHP como lenguaje para el desarrollo de la herramienta que se propone en el trabajo.

## **2.6 Sistemas Gestores de Bases de Datos**

### **2.6.1 SQL Server 2000**

SQL Server 2000 es un RDBMS (Sistema de Gestión de Bases de Datos Relacionales por sus siglas en inglés), lo que significa que permite crear, actualizar y administrar bases de datos relacionales. Una base de datos relacional es una colección de elementos de información que puede ser consultada y manipulada de varias maneras sin cambiar la organización de las tablas de la base de datos.

SQL Server 2000 soporta su propia versión del lenguaje SQL, que es conocida como Transact-SQL o TSQL, que está formada por un grupo de avanzados y poderosos comandos que pueden ser usados para crear, modificar, consultar y borrar bases de datos, tablas, triggers, restricciones etc. Está basada en el estándar ANSI SQL 92, pero le agrega un toque de flexibilidad o robustez a los comandos que la componen.

SQL Server 2000 es un potente motor de bases de datos de alto rendimiento capaz de soportar millones de registros por tabla con una interfase intuitivo y con herramientas de desarrollo integradas como Visual Studio 6.0 o .NET, además incorpora un modelo de objetos totalmente programable (SQL-DMO) con el que se puede desarrollar cualquier aplicación que manipule componentes de SQL Server, es decir, se pueden hacer aplicaciones para crear bases de datos, tablas, DTS, backups, etc., todo lo que se puede hacer desde el administrador del SQL Server y no solo en Visual C++ sino también en Visual Basic, ASP y por supuesto en .NET.

El que sea muy intuitivo en su administración o instalación no significa que sea fácil, una mala instalación, una base de datos mal creada o diseñada o una mala administración puede provocar que la aplicación tenga un rendimiento malo. También es importante el hardware,

lejos de los 64 MB mínimos que requiere el sistema es recomendable que tenga 256 o 512 para su buen funcionamiento y una cantidad suficiente de espacio en disco para que pueda trabajar con las bases de datos.

## 2.6.2 MySQL

MySQL es un sistema de gestión de bases de datos relacional, licenciado bajo la GPL de la GNU. Su diseño multihilo le permite soportar una gran carga de forma muy eficiente. MySQL fue creada por la empresa sueca MySQL AB, que mantiene el copyright del código fuente del servidor SQL, así como también de la marca.

Aunque MySQL es software libre, MySQL AB distribuye una versión comercial de MySQL, que no se diferencia de la versión libre más que en el soporte técnico que se ofrece, y la posibilidad de integrar este gestor en un software propietario, ya que de no ser así, se vulneraría la licencia GPL.

Este gestor de bases de datos es, probablemente, el gestor más usado en el mundo del software libre, debido a su gran rapidez y facilidad de uso. Esta gran aceptación es debida, en parte, a que existen infinidad de librerías y otras herramientas que permiten su uso a través de gran cantidad de lenguajes de programación, además de su fácil instalación y configuración. Las principales características de este gestor de bases de datos son las siguientes:

- Aprovecha la potencia de sistemas multiprocesador, gracias a su implementación multihilo.
- Soporta gran cantidad de tipos de datos para las columnas.
- Dispone de API's en gran cantidad de lenguajes (C, C++, Java, PHP, etc).
- Gran portabilidad entre sistemas.
- Soporta hasta 32 índices por tabla.
- Gestión de usuarios y contraseñas, manteniendo un muy buen nivel de seguridad en los datos.

MySQL surgió cómo una necesidad de un grupo de personas sobre un gestor de bases de datos rápido, por lo que sus desarrolladores fueron implementando únicamente lo que precisaban, intentando hacerlo funcionar de forma óptima. Es por ello que, aunque MySQL se

incluye en el grupo de sistemas de bases de datos relacionales, carece de algunas de sus principales características:

- Subconsultas: tal vez ésta sea una de las características que más se echan en falta, aunque gran parte de las veces que se necesitan es posible reescribirlas de manera que no sean necesarias.
- SELECT INTO TABLE: Esta característica propia de Oracle, todavía no está implementada.
- Triggers y Procedures: Se tiene pensado incluir el uso de procedimientos almacenados en la base de datos, pero no el de triggers, ya que los triggers reducen de forma significativa el rendimiento de la base de datos, incluso en aquellas consultas que no los activan.
- Integridad referencial: aunque sí que admite la declaración de llaves ajenas en la creación tablas, internamente no las trata de forma diferente al resto de campos.

### **2.6.3 Oracle**

Oracle es una herramienta cliente/servidor para la gestión de Bases de Datos. Es un producto vendido a nivel mundial, aunque la gran potencia que tiene y su elevado precio hacen que sólo se vea en empresas muy grandes y multinacionales, por norma general. En el desarrollo de páginas Web pasa lo mismo: como es un sistema muy caro no está tan extendido como otras bases de datos, por ejemplo, Access, MySQL, SQL Server, etc.

Oracle, como se había dicho previamente, se basa en la tecnología cliente/servidor, pues bien, para su utilización primero sería necesario la instalación de la herramienta servidor (Oracle 8i) y posteriormente se puede acceder a la base de datos desde otros equipos con herramientas de desarrollo como Oracle Designer y Oracle Developer, que son las herramientas básicas de programación sobre Oracle.

Para programar en Oracle se utiliza PL/SQL un lenguaje de 5ª generación, bastante potente para tratar y gestionar la base de datos, también por norma general se suele utilizar SQL al crear un formulario.

Oracle Developer es una herramienta que permite la creación de formularios en local, es decir, mediante esta herramienta se podrá crear formularios, compilarlos y ejecutarlos, pero si

se quiere que otros trabajen sobre este formulario debe ser copiado regularmente en una carpeta compartida para todos, de modo que, cuando quieran realizar un cambio, deberán copiarlo de dicha carpeta y luego volverlo a subir a la carpeta. Este sistema como se puede observar es bastante engorroso y poco fiable pues es bastante normal que las versiones se pierdan. La principal ventaja de esta herramienta es que es bastante intuitiva y dispone de un modo que permite componer el formulario, tal y como se haría en Visual Basic o en Visual C.

Los problemas anteriores quedan totalmente resueltos con Oracle Designer que es una herramienta que se conecta a la base de datos y por tanto se pueden crear los formularios en ella, todo el que vaya a trabajar se conecta mediante Oracle Designer a la aplicación que contiene todos los formularios y no hay problemas de diferentes versiones.

Pero el principal y más notable problema es la falta de un entorno visual para diseñar el formulario, es decir, aparece una estructura similar a un árbol en la cual se insertan los formularios, a la vez dentro de estos se insertan bloques o módulos, que son las estructuras que contendrán los elementos de los formularios, que pueden estar basados en tablas o no.

#### **2.6.4 PostgreSQL**

PostgreSQL es un sistema de gestión de bases de datos objeto-relacional (ORDBMS) basado en el proyecto POSTGRES, de la universidad de Berkeley. El director de este proyecto es el profesor Michael Stonebraker, y fue patrocinado por Defense Advanced Research Projects Agency (DARPA), el Army Research Office (ARO), el National Science Foundation (NSF), y ESL, Inc.

PostgreSQL es una derivación libre (OpenSource) de este proyecto, y utiliza el lenguaje SQL92/SQL99.

Fue el pionero en muchos de los conceptos existentes en el sistema objeto-relacional actual, incluido, más tarde en otros sistemas de gestión comerciales. PostgreSQL es un sistema objeto-relacional, ya que incluye características de la orientación a objetos, como puede ser la herencia, tipos de datos, funciones, restricciones, disparadores, reglas e integridad transaccional. A pesar de esto, PostgreSQL no es un sistema de gestión de bases de datos puramente orientado a objetos.

A continuación se enumeran las principales características de este gestor de bases de datos:

- Implementación del estándar SQL92/SQL99.
- Soporta distintos tipos de datos: además del soporte para los tipos base, también soporta datos de tipo fecha, monetarios, elementos gráficos, datos sobre redes (MAC, IP...), cadenas de bits, etc. También permite la creación de tipos propios.
- Incorpora una estructura de datos array.
- Incorpora funciones de diversa índole: manejo de fechas, geométricas, orientadas a operaciones con redes, etc.
- Permite la declaración de funciones propias, así como la definición de disparadores.
- Soporta el uso de índices, reglas y vistas.
- Incluye herencia entre tablas (aunque no entre objetos, ya que no existen), por lo que a este gestor de bases de datos se le incluye entre los gestores objeto-relacionales.
- Permite la gestión de diferentes usuarios, como también los permisos asignados a cada uno de ellos.

PostgreSQL es un magnífico gestor de bases de datos, capaz de competir con muchos gestores comerciales, aunque carezca de un conjunto de herramientas que permitan una fácil gestión de los usuarios y de las bases de datos que contenga el sistema. Por otro lado, la velocidad de respuesta que ofrece este gestor con bases de datos relativamente pequeñas puede parecer un poco deficiente, aunque esta misma velocidad la mantiene al gestionar bases de datos realmente grandes, cosa que resulta loable.

### **2.6.5 Fundamentación del Sistema Gestor Seleccionado**

Luego de analizar con profundidad las características de los sistemas gestores más usados a nivel mundial, y teniendo en cuenta el tipo de herramienta que se propone en el trabajo, se escoge PostgreSQL como sistema gestor de base de datos a utilizar debido esencialmente a que:

- Posee una gran escalabilidad. Es capaz de ajustarse al número de CPUs y a la cantidad de memoria que posee el sistema de forma óptima, haciéndole capaz de soportar una mayor cantidad de peticiones simultáneas de manera correcta.



- Implementa el uso de rollback's, subconsultas y transacciones, haciendo su funcionamiento mucho más eficaz, y ofreciendo soluciones en campos en las que otros gestores como MySQL no podría.
- Tiene la capacidad de comprobar la integridad referencial, así como también la de almacenar procedimientos en la propia base de datos, equiparándolo con los gestores de bases de datos de alto nivel, como puede ser Oracle.

## ***2.7 Herramientas Propuestas para el Desarrollo de la Aplicación.***

### **2.7.1 Visual Paradigm**

Las Herramientas CASE pueden ser definidas como un conjunto de programas y ayudas que dan asistencia a los analistas, ingenieros de software y desarrolladores, durante todos los pasos del Ciclo de Vida de de desarrollo de software.

El desarrollo de un nuevo software requiere que las tareas sean organizadas y completadas de forma correcta y eficiente. Las Herramientas CASE (Ingeniería de Software Asistida por Ordenador o CASE por sus siglas en inglés) fueron desarrolladas para automatizar esos procesos y facilitar las tareas de coordinación de los eventos que necesitan ser mejorados en el ciclo de desarrollo de software. Permiten incrementar la productividad y el control de calidad en cualquier proceso de elaboración de software, ya que transforman el desarrollo de software en un proceso automatizado.

Estas herramientas proveen muchos beneficios en todas las etapas del proceso de desarrollo de software, como por ejemplo:

- Permitir la aplicación práctica de metodologías, lo que resulta complicado sin emplear herramientas informáticas.
- Verificar el uso de todos los elementos en el sistema diseñado.
- Automatizar la elaboración de diagramas.
- Simplificar el mantenimiento del software.
- Mejorar y estandarizar la documentación.
- Aumentar la portabilidad de las aplicaciones.
- Facilitar la reutilización de componentes de software.
- Ayudar en el diseño de bases de datos.

Visual Paradigm3 es una suite completa de herramientas CASE que da soporte al modelado visual con UML 2.0 ofreciendo distintas perspectivas del sistema. Independiente de la plataforma y dotada de una buena cantidad de productos o módulos para facilitar el trabajo durante la confección de un software así como garantizar la calidad del producto final.

Sus principales características son:

- Es profesional: brinda la posibilidad de crear un conjunto bastante amplio de artefactos utilizados con mucha frecuencia durante la confección de un Software. Todos estos, cumpliendo con el Standard UML 2.0.
- Es amigable: puede ser utilizado en varios idiomas, sus componentes se encuentran relacionados, por lo que se hace muy fácil la creación de cualquier tipo de diagrama, ya que cada componente utilizado en el diagrama que se esté creando, sugiere nuevos posibles componentes a utilizar, por lo que ya no es necesario localizarlos en la barra donde pueden aparecer un número grande de componentes.
- Brinda un número considerable de estereotipos a utilizar, lo que permite un mayor entendimiento de los diagramas.
- Facilidades para redactar especificaciones de casos de uso: es posible crear plantillas para las especificaciones de casos de uso y describirlos, por lo que no se necesita de una herramienta externa como editor de texto.
- Generación de código e ingeniería inversa: brinda la posibilidad de generar código a partir de los diagramas, para plataformas como .Net, Java y PHP, así como obtener diagramas a partir del código.
- Integración con distintos Ambientes de Desarrollo Integrados (IDE): se integra fácilmente con varios IDEs, entre ellos el de Visual Studio y el Eclipse.
- Interoperabilidad con otras aplicaciones: brinda la posibilidad de intercambiar información mediante la importación y exportación de ficheros con aplicaciones como por ejemplo Visio y Rational Rose. Además permite importar y exportar XML y XMI.
- Generación de código ORM: permite generar a partir de un Diagrama de Entidad Relación una Base de Datos Relacional y el código necesario para acceder a esta base de datos utilizando Java, PHP, C# o Enterprise Object Framework.
- Generación de documentación: brinda la posibilidad de documentar todo el trabajo sin necesidad de utilizar herramientas externas.

- Disponibilidad en múltiples plataformas: Microsoft Windows (98, 2000, XP, o Vista), Linux, Mac OS X, Solaris o Java.

### 2.7.2 Zend Studio

Zend Studio combina la probada tecnología de Zend con las herramientas para desarrolladores en PHP (PDT por sus siglas en inglés) de Eclipse para crear un, muy poderoso, IDE para el desarrollo de aplicaciones WEB.

El programa entero está escrito en Java, lo que a veces supone que no funcione tan rápido como otras aplicaciones de uso diario. Sin embargo, esto ha permitido a Zend lanzar con relativa facilidad y rapidez versiones del producto para Windows, Linux y MacOS.

Zend Studio consta de dos partes en las que se dividen las funcionalidades de parte del cliente y las del servidor. Las dos partes se instalan por separado, la del cliente contiene el interfaz de edición y la ayuda. Permite, además, hacer depuraciones simples de scripts, aunque para disfrutar de toda la potencia de la herramienta de depuración habrá que disponer de la parte del servidor, que instala Apache y el módulo PHP o, en caso de que estén instalados, los configura para trabajar juntos en depuración.

**El editor** es la parte del programa que permite escribir los scripts es bastante útil para la programación en PHP. La interfaz está compuesta por varias partes, en las que encontramos un explorador de archivos, una ventana de depuración, los menús y otra para mostrar el código de las páginas.

Lo más destacable es que contiene una ayuda contextual con todas las librerías de funciones del lenguaje que asiste en todo momento ofreciendo nombres de las funciones y parámetros que deben recibir. Aunque esta ayuda contextual no solo se queda en las funciones definidas en el lenguaje, sino que también reporta ayudas con las funciones que vaya creando el desarrollador, incluso en páginas que se tengan incluidas con la función include().

**El gestor de proyectos** posee una barra a la izquierda, que permite navegar los archivos de nuestro ordenador, también dispone de herramientas para gestionar los proyectos, muy útiles para mejorar la productividad en la programación. Los proyectos permiten guardar mucha más

información al programa sobre los archivos, discos, servidores, etc. que se gestionen las aplicaciones PHP.

Zend Studio implementa además unas interesantes opciones para trabajar en grupo, al integrar el sistema de trabajo conocido como CVS (Sistema de Versiones Concurrentes por sus siglas en inglés).

No cabe duda que más de una vez los programadores de PHP se han visto en un duro problema por no encontrar un error en algún script que está haciendo que la aplicación devuelva resultados inesperados. En estos casos lo que se suele hacer es escribir el contenido de diversas variables en la página Web y rezar para que aparezca algún indicio del lugar donde está el error.

Para evitar lo engorroso de este proceso Zend Studio dispone de una **Herramienta de Depuración** muy interesante que facilita, además de la propia depuración, el debug de la aplicación. Gracias a ella se pueden ejecutar las páginas y conocer en todo momento el contenido de las variables de la aplicación y las variables del entorno como las cookies, las recibidas por formulario o en la sesión. Es posible colocar puntos de parada de los scripts y realizar las acciones típicas de depuración.

Además de la ventana para visualizar el contenido de las variables, dispone de otras donde muestra la salida del script según se va generando, y otra donde se pueden ver las alertas y errores. Las posibilidades se completan con distintos tipos de depuración, en local, en remoto o a partir de una URL.

### **2.7.3 PgAdmin III**

PgAdmin III es una aplicación gráfica para gestionar el gestor de bases de datos PostgreSQL, siendo la más completa y popular con licencia Open Source. Está escrita en C++ usando la librería gráfica multiplataforma wxWidgets, lo que permite que se pueda usar en Linux, FreeBSD, Solaris, Mac OS X y Windows. Es capaz de gestionar versiones de PostgreSQL 7.3 y anteriores, así como versiones comerciales de PostgreSQL como Pervasive Postgres, EnterpriseDB, Mammoth Replicator y SRA PowerGres.

PgAdmin III está diseñado para responder a las necesidades de todos los usuarios, desde escribir consultas SQL simples hasta desarrollar bases de datos complejas. El interfaz gráfico soporta todas las características de PostgreSQL y facilita enormemente la administración. La aplicación también incluye un editor SQL con resaltado de sintaxis, un editor de código de la parte del servidor, un agente para lanzar scripts programados, soporte para el motor de replicación Slony-I y mucho más. La conexión al servidor puede hacerse mediante conexión TCP/IP o Unix Domain Sockets (en plataformas \*nix), y puede encriptarse mediante SSL para mayor seguridad.

PgAdmin III fue desarrollado por una comunidad de expertos en PostgreSQL de todo el mundo, esta disponible en más de una docena de lenguajes. Es un software gratis liberado bajo "Artistic License".

## **2.8 Conclusiones**

En este capítulo se hizo un análisis de las tendencias tecnológicas actuales, las principales características, de metodologías, lenguajes de programación y modelado, herramientas etc., fueron descritas. Se llega a la conclusión de que la propuesta hará uso de la Metodología RUP para su desarrollo, se utilizará la arquitectura en 3 capas y UML como lenguaje de modelado, Visual Paradigm como herramienta CASE, PHP como lenguaje de programación y PostgreSQL como sistema gestor de bases de datos, Zend Studio como IDE de desarrollo y pgAdmin III como plataforma para la gestión de la base de datos del sistema. Teniendo en cuenta las características de las tendencias seleccionadas así como la política de migración hacia software libre de nuestro país, se puede realizar la afirmación de que el conjunto de tendencias seleccionadas es el óptimo para el desarrollo de la herramienta que propone el trabajo.

## Capítulo 3 “Presentación de la solución propuesta”

### 3.1 Introducción

En el siguiente capítulo se describe la solución que se propone como respuesta a la situación problemática. Se desarrolla un modelo de negocio para lograr el entendimiento de los procesos de este. Además se enumeran los requisitos del sistema (funcionales y no funcionales) con el objetivo de ofrecer una concepción general de lo que, el sistema, debe cumplir. Se hace uso del diagrama de casos de uso para describir las relaciones de los actores con el sistema.

### 3.2 Modelo de Negocio

#### 3.2.1 Actores y trabajadores del negocio

Actores del Negocio	Justificación
Director del Proyecto	Es el que inicia la ejecución de todos los procesos que se incluyen dentro de la gestión de las áreas específicas del proyecto.

Tabla 1: Actores del Negocio.

Trabajadores del Negocio	Justificación
Gestor del Tiempo.	Es el encargado de ejecutar los procesos contenidos dentro de la gestión del tiempo del proyecto.
Gestor del Costo.	Es el encargado de ejecutar los procesos contenidos dentro de la gestión del costo del proyecto.
Gestor de los Riesgos.	Es el encargado de ejecutar los procesos contenidos dentro de la gestión de los riesgos del proyecto.
Gestor del Alcance.	Es el encargado de ejecutar los procesos

	contenidos dentro de la gestión del alcance del proyecto.
Gestor de la Calidad.	Es el encargado de ejecutar los procesos contenidos dentro de la gestión de la calidad del proyecto.
Gestor de los Recursos Humanos.	Es el encargado de ejecutar los procesos contenidos dentro de la gestión de la calidad del proyecto.
Gestor de las Comunicaciones.	Es el encargado de ejecutar los procesos contenidos dentro de la gestión de las comunicaciones del proyecto.
Gestor del Aprovisionamiento.	Es el encargado de ejecutar los procesos contenidos dentro de la gestión del aprovisionamiento del proyecto.

**Tabla 2: Trabajadores del Negocio**

### **3.2.2 Procesos de Negocio.**

La Gestión de Proyectos tiene como objetivo lograr la creación de un producto, servicio o resultado de calidad, dentro del rango de tiempo, costo y alcance, previamente definido.

En este negocio presenta un actor que sería el Equipo de Dirección del Proyecto, que es el encargado de verificar que se ejecuten correctamente los procesos que componen al proyecto.

El negocio presenta además 8 trabajadores que serían los Equipos de Gestión del tiempo, costo, alcance, riesgos, calidad, recursos humanos, comunicaciones y aprovisionamiento, que son los encargados de ejecutar cada uno de los procesos para que el proyecto termine con los el resultado esperado.

Ahora, como ya se había dejado de manera implícita, en este capítulo, los procesos se agrupan dentro de varias áreas de gestión:

#### *1. Gestión del tiempo.*

En esta área de gestión se definirán cada una de las actividades que deben ser completadas para la terminación del proyecto, se realiza un secuenciamiento de dichas actividades, se hace una estimación de los recursos necesarios para ejecutar dichas actividades, se

desarrolla el Schedule del proyecto y se gestionan los cambios a realizar en cualquiera de los procesos previamente enumerados.

## 2. *Gestión del Costo.*

En esta área de gestión se realizará una estimación del costo de ejecución de cada una de las actividades del proyecto, se elabora el presupuesto del proyecto, y se realiza el control de los costos que consiste en ejercer presión sobre los factores que podrían causar una variación de los costos previamente estimados así como la gestión de los cambios que habría que hacer en la ejecución de los procesos previamente enumerados.

## 3. *Gestión del Alcance.*

En esta área de gestión se realizará una planificación de lo que será el alcance del proyecto donde se prevén las políticas referente a la administración del mismo, luego se define el alcance del proyecto, se crea el WBS(Work Breakdown Structure) donde se divide al proyecto en subproyectos más pequeños y funcionales en dependencia de la complejidad de dicho proyecto, se verifica el alcance del proyecto llegándose a un consenso final con los stakeholders de cuál será el alcance definitivo, se gestionan los cambios en la ejecución de cualquiera de los procesos previamente enumerados.

## 4. *Gestión de los Riesgos.*

En esta área de gestión se realizará una planificación de los riesgos definiéndose las políticas de como serán tratados, se identifican los riesgos del proyecto, se realiza un análisis cualitativos de los riesgos, se realiza un análisis cuantitativo de los riesgos, se elabora una planificación de las respuestas que serán dadas si algún riesgo llega a cumplirse, se monitorean y controlan los riesgos los que consiste en prever si algún riesgo está por cumplirse o si ha surgido algún riesgo nuevo y además se gestionan los cambios en la ejecución de cualquiera de los procesos enumerados previamente.

## 5. *Gestión de la Calidad.*

En esta área de gestión se realiza una planificación de la calidad definiéndose las políticas de calidad que usará el proyecto, se realiza el aseguramiento de la calidad que consiste en garantizar que el proyecto hace lo correcto para obtener un resultado satisfactorio, se controla la calidad que consiste en inspeccionar los entregables del proyecto para ver si cumplen con los requisitos de calidad previamente definidos y además se gestionan los cambios en cualquiera de los procesos previamente enumerados.

## 6. *Gestión de los Recursos Humanos.*

En esta área de gestión se realiza una planificación de los recursos humanos donde se definen las políticas en cuanto al tratamiento de los RRHH, se realiza la adquisición del equipo



de proyecto, se desarrolla el equipo de proyecto, se gestiona el equipo de proyecto que consiste en darle seguimiento a cada uno de los miembros del equipo de proyecto así como gestionar los cambios a realizar en cualquiera de los procesos previamente enumerados.

#### 7. *Gestión de las Comunicaciones.*

En esta área de gestión se realiza una planificación de las comunicaciones del proyecto donde se definen las políticas de disponibilidad de la información, se distribuye la información donde se asegura la disponibilidad de esta, se realiza el reporte de rendimiento, se gestionan los stakeholders para solucionar problemas o cualquier cosa preocupante para ellos.

#### 8. *Gestión del Aprovechamiento.*

En esta área de gestión se realiza una planificación que contiene las políticas a seguir en cuanto a las compras y adquisiciones que realizará el proyecto, se hace una planificación de los contratos ha firmar, se solicita la respuesta a los posibles vendedores, se seleccionan los vendedores definitivos dentro de los que respondieron positivamente, se administran los contratos del proyectos, se gestionan los cambios de cualquiera de los procesos enumerados previamente, se realiza el cierre de los compromisos contractuales del proyecto.

### 3.2.3 Diagrama de Casos de Uso del Negocio

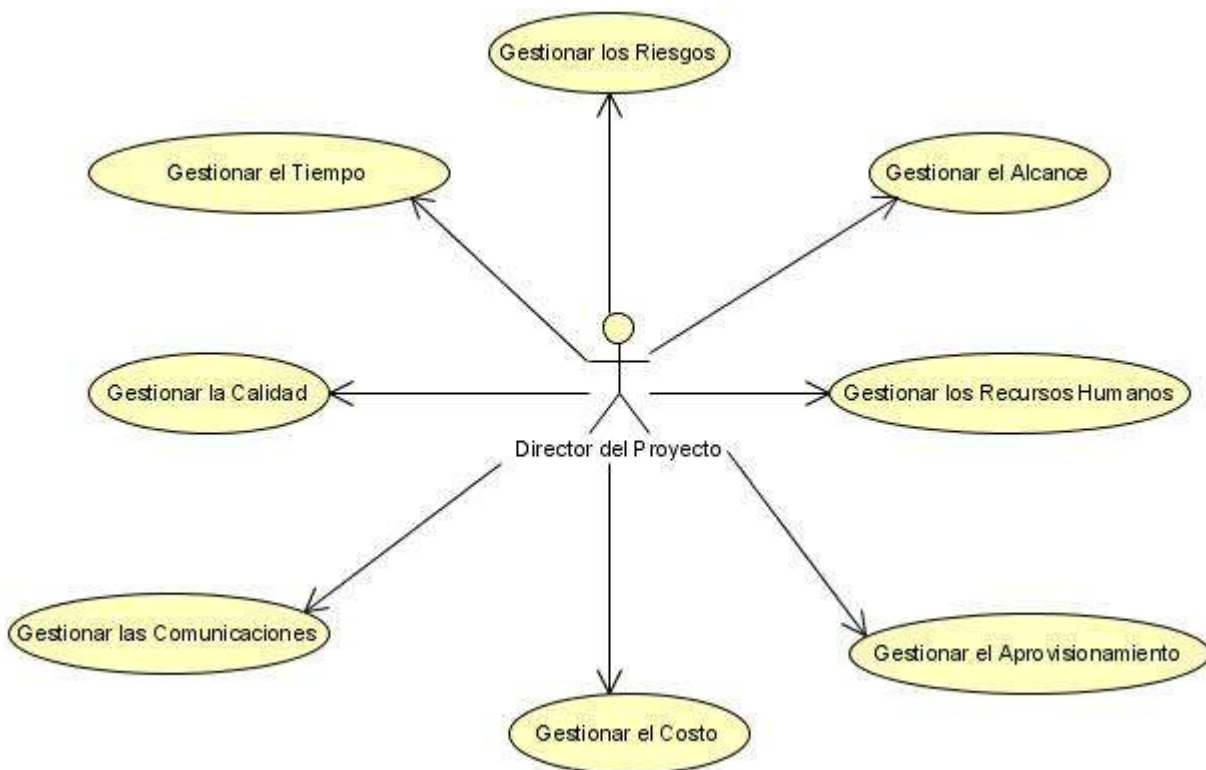


Figura 1: Diagrama de Casos de uso del Negocio.

### 3.2.4 Diagramas de Actividades de los Casos de Uso de Negocio

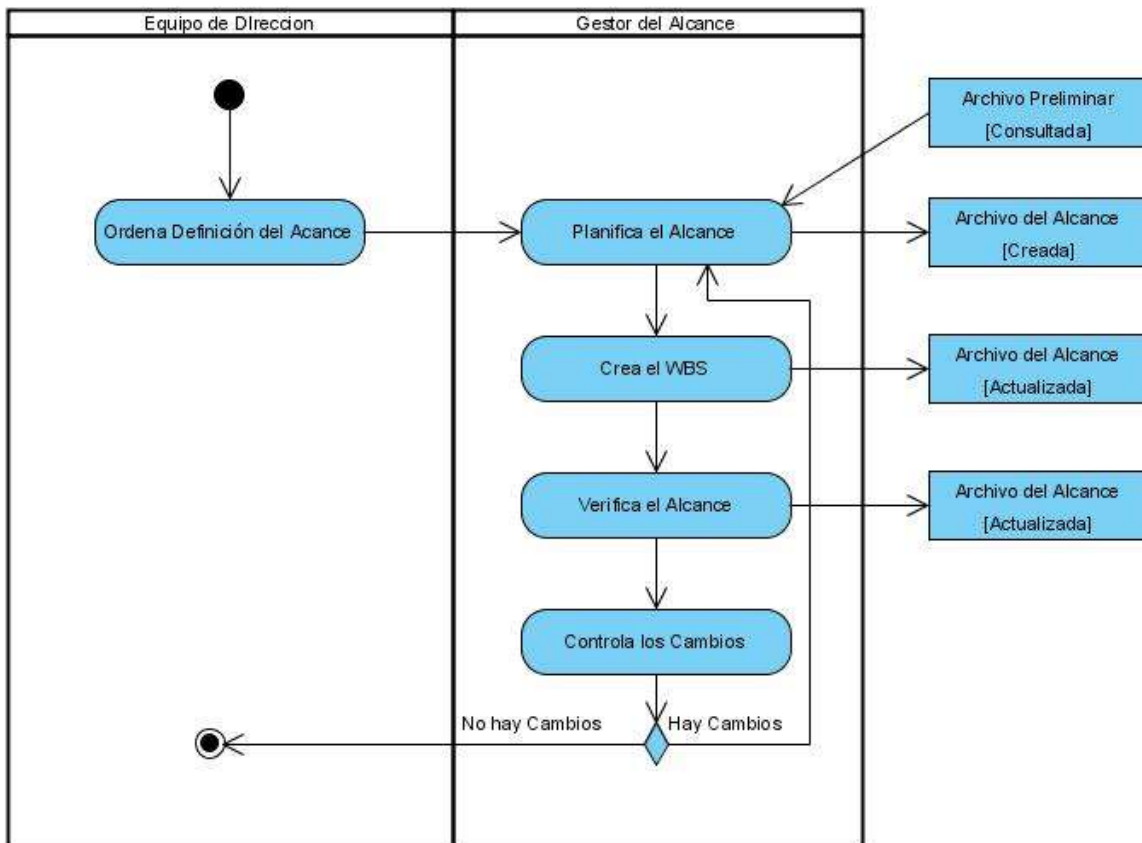


Figura 2: Diagrama de Actividades CUN Gestionar el Alcance.

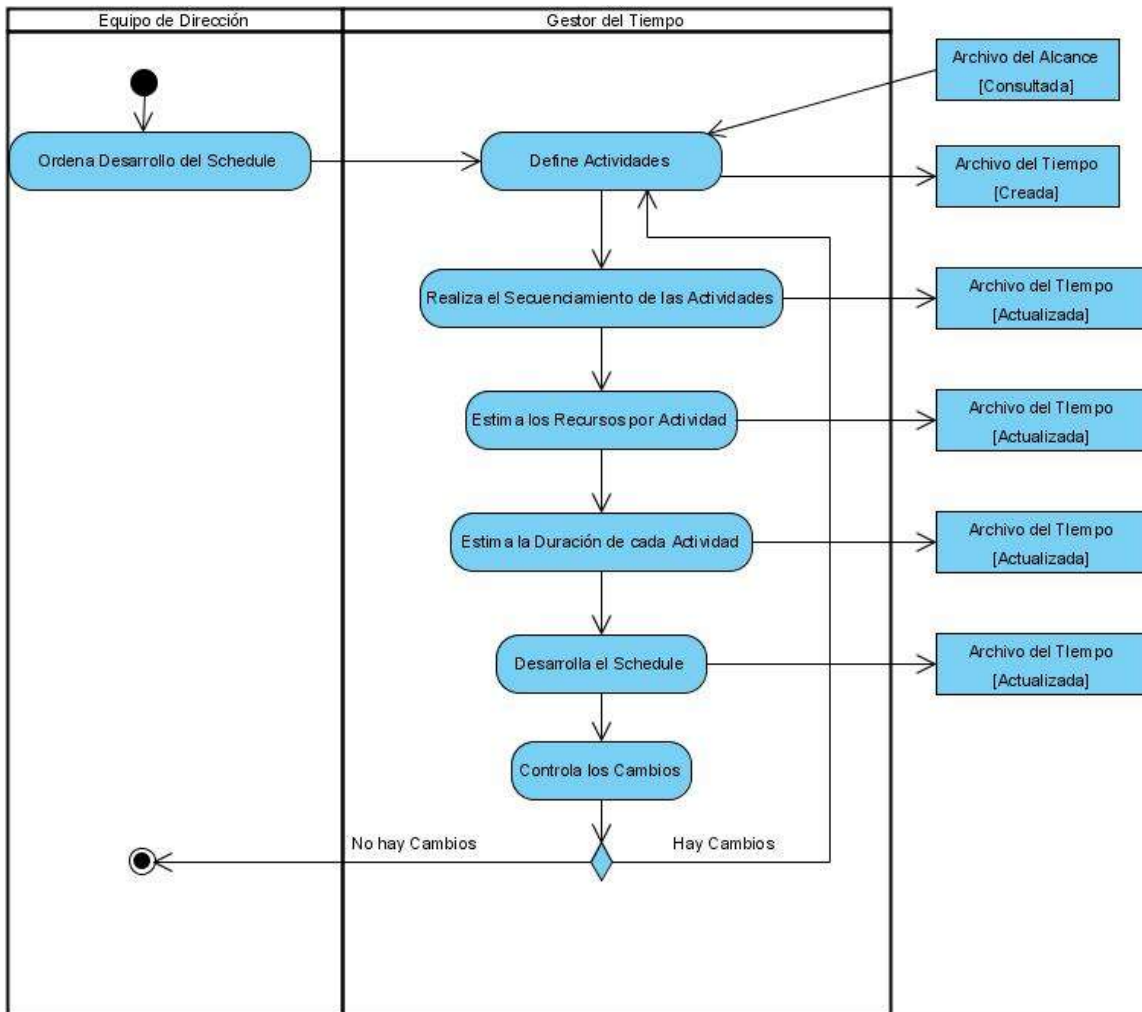


Figura 3: Diagrama de Actividades CUN Gestionar el Tiempo.

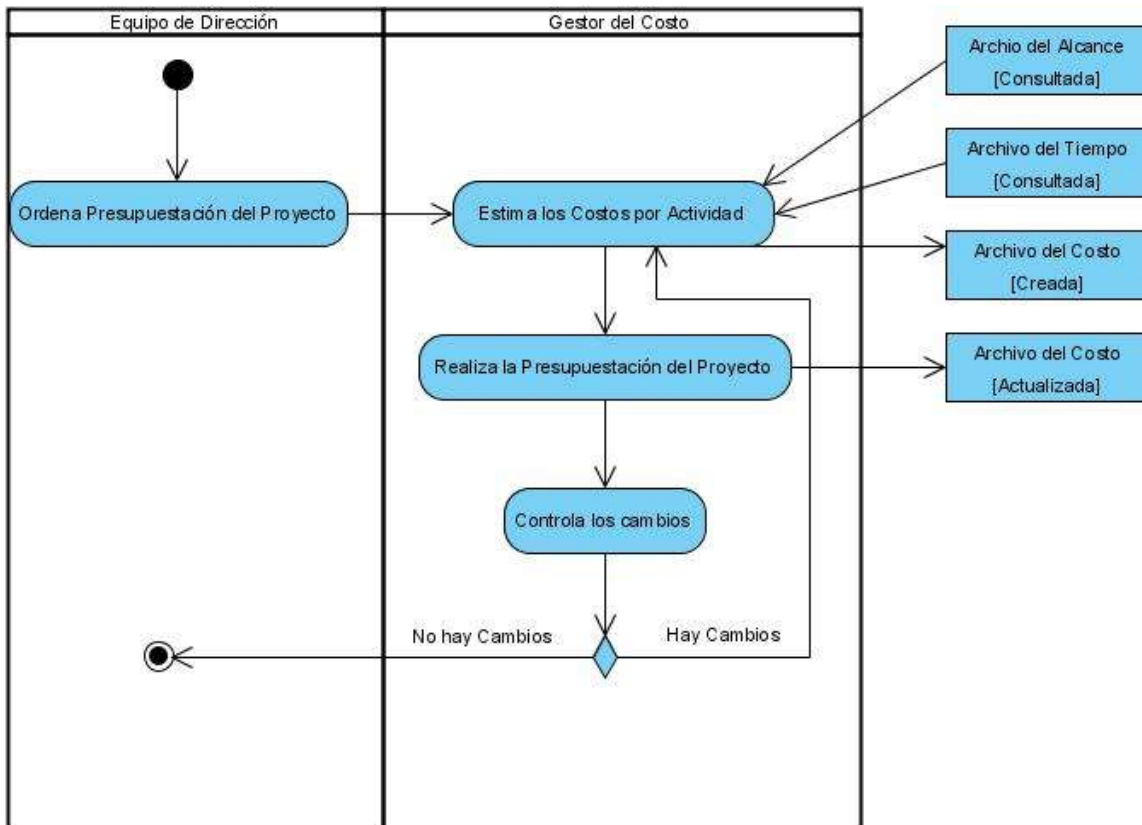


Figura 4: Diagrama de Actividades CUN Gestionar el Costo.

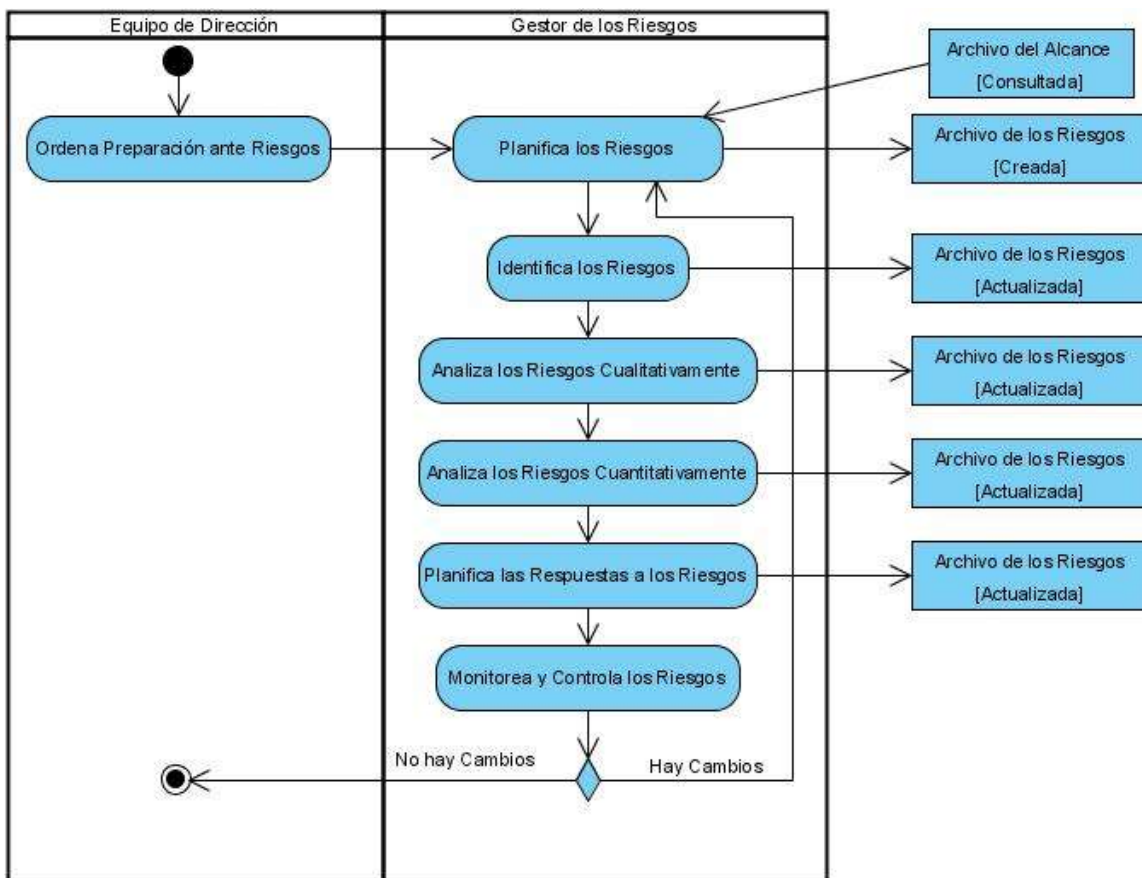


Figura 5: Diagrama de Actividades CUN Gestionar los Riesgos.

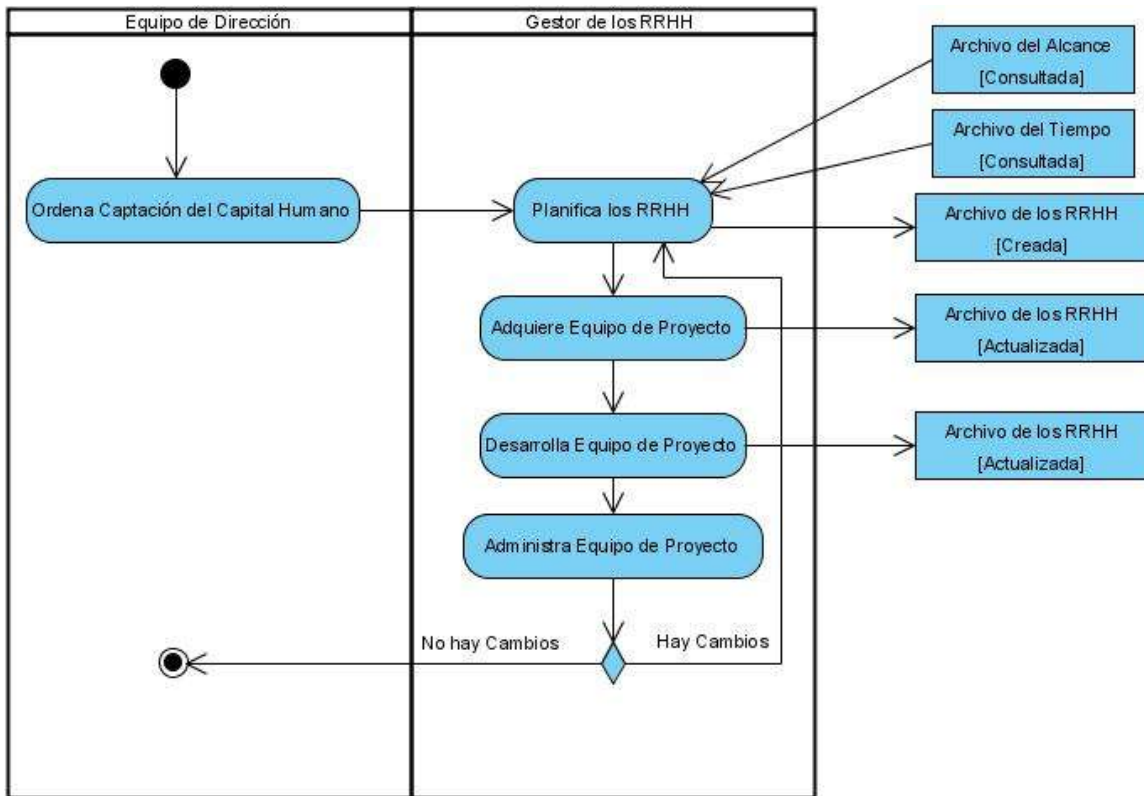


Figura 6: Diagrama de Actividades CUN Gestionar los RRHH.

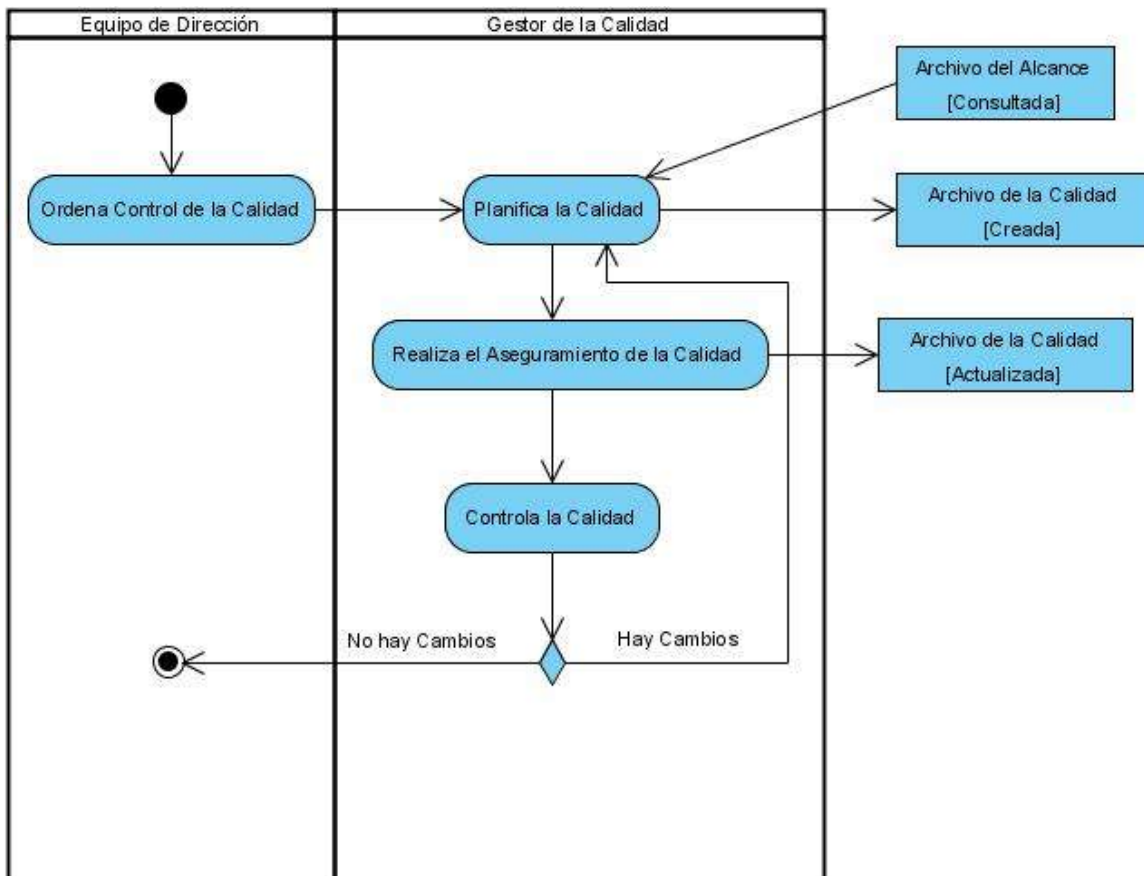
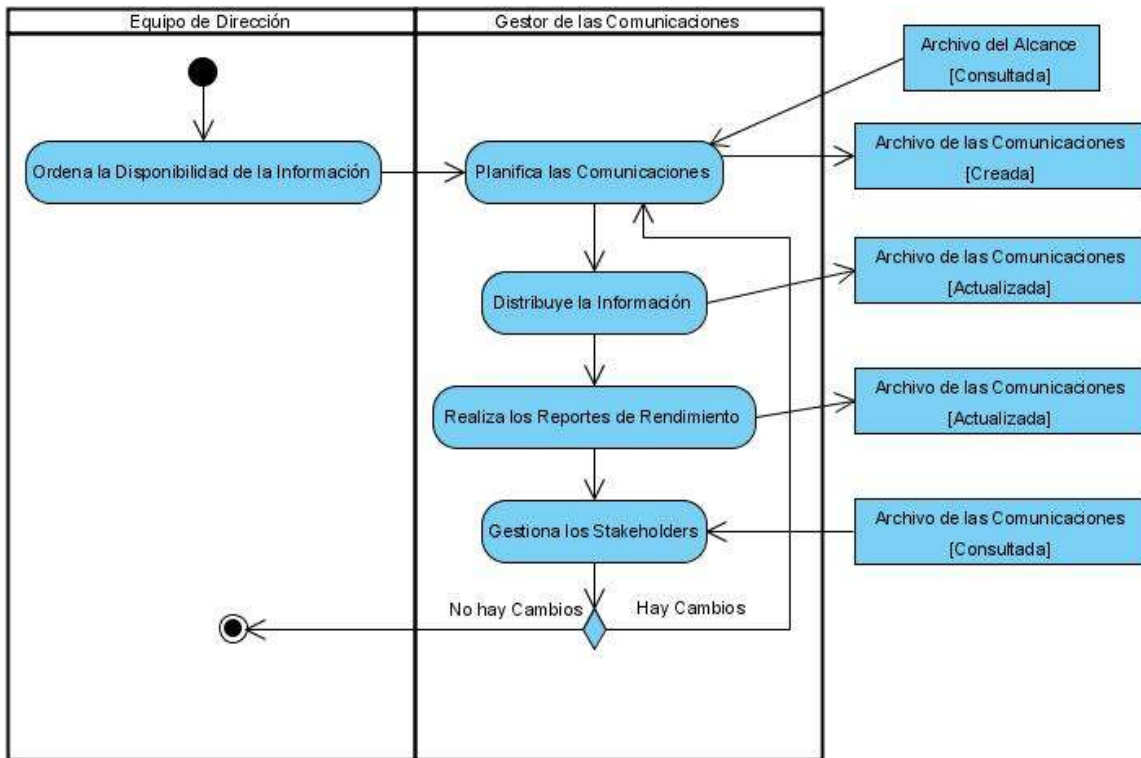
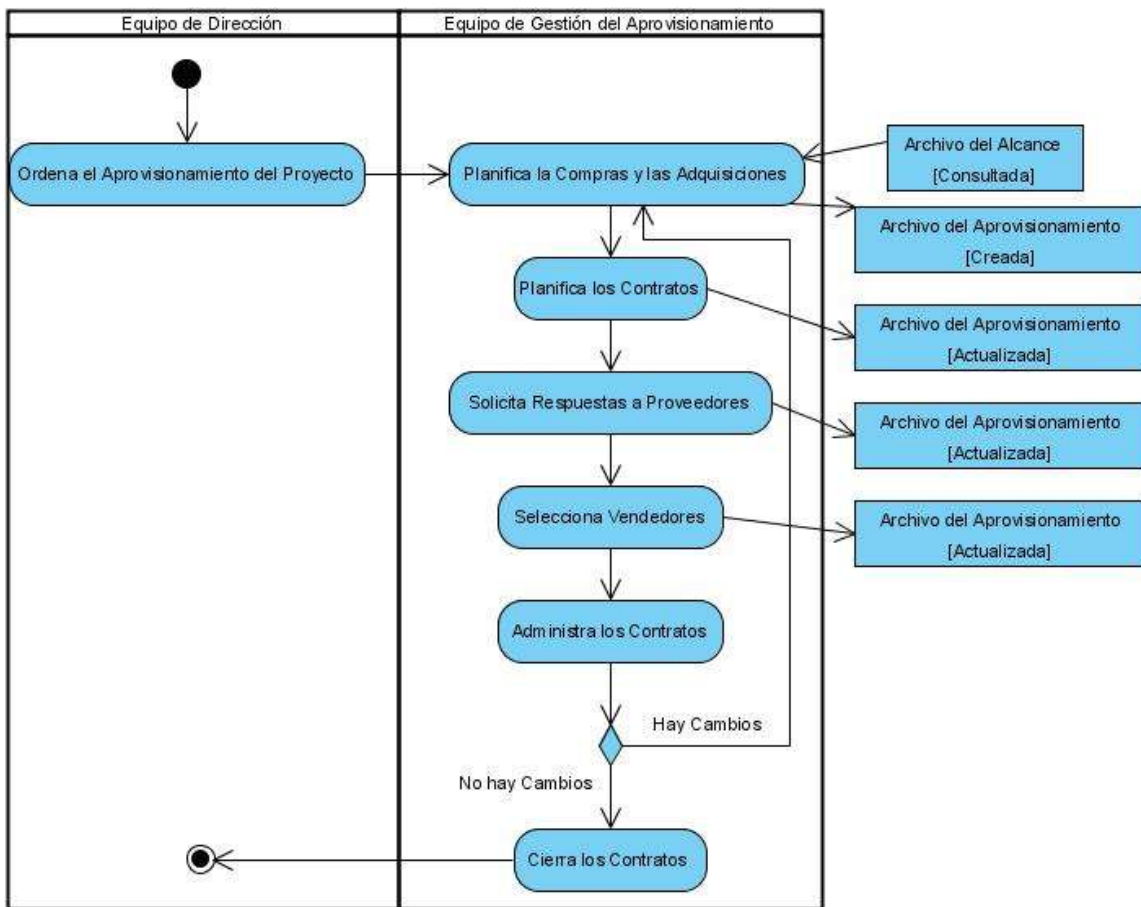


Figura 7: Diagrama de Actividades CUN Gestionar la Calidad.



**Figura 8: Diagrama de Actividades CUN Gestionar las Comunicaciones.**



**Figura 9: Diagrama de Actividades CUN Gestionar el Aprovisionamiento.**

### **3.3 *Requerimientos del Software.***

#### **3.3.1 *Requerimientos Funcionales.***

De acuerdo a los objetivos del trabajo el sistema debe ser capaz de:

**R1.** Almacenar y mostrar la información acerca de los Documentos del Proyecto (ver glosario de términos para Documentos del Proyecto).

**R1.1** Almacenar y mostrar la información acerca de los Documentos del Tiempo.

**R1.1.1** Adicionar un nuevo documento permitiendo el almacenamiento de su información.

**R1.1.2** Adicionar la actualización de un documento previamente adicionado.

**R1.1.3** Mostrar los Documentos del Tiempo Previamente Adicionados.

**R1.1.4** Mostrar la traza de un documento del proyecto previamente adicionado.

**R1.2** Almacenar y mostrar la información acerca de los Documentos del Costo.

**R1.2.1** Adicionar un nuevo documento permitiendo el almacenamiento de su información.

**R1.2.2** Adicionar la actualización de un documento previamente adicionado.

**R1.2.3** Mostrar los Documentos del Costo Previamente Adicionados.

**R1.2.4** Mostrar la traza de un documento del proyecto previamente adicionado.

**R1.3** Almacenar y mostrar la información acerca de los Documentos de los Riesgos.

**R1.3.1** Adicionar un nuevo documento permitiendo el almacenamiento de su información.

**R1.3.2** Adicionar la actualización de un documento previamente adicionado.

**R1.3.3** Mostrar los Documentos de los Riesgos Previamente Adicionados.

**R1.3.4** Mostrar la traza de un documento del proyecto previamente adicionado.

**R1.4** Almacenar y mostrar la información acerca de los Documentos del Alcance.

**R1.4.1** Adicionar un nuevo documento permitiendo el almacenamiento de su información.

**R1.4.2** Adicionar la actualización de un documento previamente adicionado.

**R1.4.3** Mostrar los Documentos del Alcance Previamente Adicionados.

**R1.4.4** Mostrar la traza de un documento del proyecto previamente adicionado.

**R1.5** Almacenar y mostrar la información acerca de los Documentos de la Calidad.

**R1.5.1** Adicionar un nuevo documento permitiendo el almacenamiento de su información.

**R1.5.2** Adicionar la actualización de un documento previamente adicionado.

**R1.5.3** Mostrar los Documentos de la Calidad Previamente Adicionados.

**R1.5.4** Mostrar la traza de un documento del proyecto previamente adicionado.

**R1.6** Almacenar y mostrar la información acerca de los Documentos de los Recursos Humanos.

**R1.6.1** Adicionar un nuevo documento permitiendo el almacenamiento de su información.

- R1.6.2** Adicionar la actualización de un documento previamente adicionado.
- R1.6.3** Mostrar los Documentos de los Recursos Humanos Tiempo Previamente Adicionados.
- R1.6.4** Mostrar la traza de un documento del proyecto previamente adicionado.
- R1.7** Almacenar y mostrar la información acerca de los Documentos de las Comunicaciones.
  - R1.7.1** Adicionar un nuevo documento permitiendo el almacenamiento de su información.
  - R1.7.2** Adicionar la actualización de un documento previamente adicionado.
  - R1.7.3** Mostrar los Documentos de las Comunicaciones Previamente Adicionados.
  - R1.7.4** Mostrar la traza de un documento del proyecto previamente adicionado.
- R1.8** Almacenar y mostrar la información acerca de los Documentos del Aprovisionamiento.
  - R1.8.1** Adicionar un nuevo documento permitiendo el almacenamiento de su información.
  - R1.8.2** Adicionar la actualización de un documento previamente adicionado.
  - R1.8.3** Mostrar los Documentos del Aprovisionamiento Previamente Adicionados.
  - R1.8.4** Mostrar la traza de un documento del proyecto previamente adicionado.
  
- R2.** Almacenar y mostrar la información acerca de las Actividades del Proyecto.
  - R2.1** Adicionar una nueva actividad permitiendo el almacenamiento de su información.
    - R2.1.1** Almacenar y mostrar la información acerca de los Recursos de una determinada Actividad.
      - R2.1.1.1** Adicionar un nuevo recurso permitiendo el almacenamiento de su información.
      - R2.1.1.2** Modificar la información almacenada de un Recurso previamente adicionado.
      - R2.1.1.3** Eliminar la información almacenada de un Recurso previamente adicionado.
    - R2.2** Modificar la información previamente almacenada de una Actividad.
    - R2.3** Eliminar la información previamente almacenada de una Actividad.
    - R2.4** Mostrar el Schedule del Proyecto.
      - R2.4.1** Generar Diagrama de Gantt con la información de las actividades previamente almacenadas.
  
- R3.** Almacenar y mostrar la información acerca de los Costos del Proyecto.
  - R3.1** Asignarle el Costo a un Recurso determinado.
  - R3.2** Mostrar un Informe detallado de los Costos del proyecto ordenados por actividad y por recursos.
  
- R4.** Almacenar y mostrar la información acerca de los Riesgos de un Proyecto.
  - R4.1** Adicionar un Nuevo Riesgo permitiendo el almacenamiento de su información.



**R4.2** Modificar la información previamente almacenada de un Riesgo.

**R4.3** Eliminar la información previamente almacenada de un Riesgo.

**R5.** Permitir al usuario identificarse ante el sistema con el objetivo de que este pueda asignarle los permisos para que pueda acceder solo a las secciones del sistema a las que esta autorizado.

### **3.3.2 Requerimientos No Funcionales.**

#### **Apariencia e Interfaz.**

La interfaz del sistema se verá a través de un navegador Web. La aplicación debe contar con un diseño sencillo y explícito que permita al usuario interactuar con el sistema sin necesitar un entrenamiento profundo, debe ser un diseño sobrio y formal con toda la seriedad que una aplicación de este tipo requiere.

#### **Rendimiento.**

El rendimiento del sistema debe depender solo de la velocidad de transmisión de la red así como de la capacidad de procesamiento del servidor de bases de datos.

#### **Soporte.**

##### *Servidor:*

Se debe contar con un servidor Web que soporte PHP 5 o superior, además se debe contar con un sistema gestor de bases de datos que de PostGreSQL 5 o superior.

##### *Cliente:*

Se requiere la instalación en cada estación cliente un navegador Web que soporte JavaScript.

#### **Seguridad.**

Se deben definir distintos roles de acuerdo a la actividad que le corresponda hacer al usuario en el sistema y se debe autenticar a cada usuario antes de que comience a interactuar con el sistema.

Se debe cuestionar al usuario cada vez que vaya a efectuar una acción irreversible (cuando se modifica o elimina algo).

Se utilizarán mecanismos de encriptación para evitar que la información viaje en forma de texto plano.

La aplicación debe estar diseñada de forma modular para que cada usuario pueda y tenga que acceder solo a la sección donde le corresponde trabajar.

### **Portabilidad.**

El sistema debe ser capaz de correr sobre los sistemas operativos Windows de Microsoft y las distribuciones más populares de Linux como Suse, Ubuntu, Fedora, Red Hat, Gentoo y Debian.

## **3.4 Descripción del Sistema Propuesto.**

### **3.4.1 Descripción de los actores.**

<b>Actores del Sistema</b>	<b>Justificación</b>
Gestor del Tiempo.	Personas encargadas de gestionar las actividades y el Schedule del proyecto así como todos los documentos relacionados con dicha gestión.
Gestor del Costo.	Personas encargadas de gestionar los costos de los recursos del proyecto así como todos los documentos relacionados con dicha gestión.
Gestor de los Riesgos.	Personas encargadas de gestionar los riesgos del proyecto así como todos los documentos relacionados con dicha gestión.
Gestor del Alcance.	Personas encargadas de gestionar los documentos relacionados con la gestión del alcance del proyecto.
Gestor de la Calidad.	Personas encargadas de gestionar los documentos relacionados con la gestión de la calidad del proyecto.
Gestor de los Recursos	Personas encargadas de gestionar los

Humanos.	documentos relacionados con la gestión de los recursos humanos del proyecto.
Gestor de las Comunicaciones.	Personas encargadas de gestionar los documentos relacionados con la gestión de las comunicaciones del proyecto.
Gestor del Aprovisionamiento.	Personas encargadas de gestionar los documentos relacionados con la gestión del aprovisionamiento del proyecto.
Supervisor	Persona que podrá acceder a toda la información del proyecto.
Usuario.	Todas las personas autorizadas a interactuar con la aplicación, es una generalización de todos los actores descritos previamente.
Gestor de Actividades	Personas autorizadas a manejar la información de las actividades del proyecto, es una generalización de los actores Equipo de Gestión del Tiempo y Supervisor.
Gestor de Costos	Personas autorizadas a manejar la información de los costos del proyecto, es una generalización de los actores Equipo de Gestión del Costo y Supervisor.
Gestor de Riesgos	Personas autorizadas a manejar la información de los Riesgos del Proyecto, es una generalización de los actores Equipo de Gestión de los Riesgos y Supervisor.

**Tabla 3: Actores del Sistema.**

### 3.4.2 Casos de Uso del Sistema.

#### 3.4.2.1 Subsistema Autenticación.

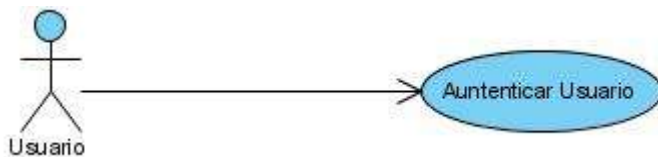


Figura 10: Diagrama de Casos de Uso Subsistema Autenticación.

<b>Nombre del CU</b>	<b>Auntenticar Usuario</b>
<b>Actores</b>	Usuario (inicia)
<b>Objetivo</b>	Brindar los permisos requeridos en dependencia del usuario que sea.
<b>Resumen</b>	El caso de uso se inicia cuando el usuario inserta sus datos al sistema, el sistema los verifica y le asigna al usuario los permisos correspondientes, en caso de ser incorrectos se le niega el acceso a la aplicación mostrando un mensaje de error.
<b>Referencias</b>	<b>R5</b>
<b>Precondiciones</b>	Debe estar registrado en la base de datos de la aplicación.
<b>Poscondiciones</b>	El sistema otorga los permisos correspondientes a cada usuario.
<b>Curso Normal de los Eventos</b>	
<b>Acciones del Actor</b>	<b>Respuesta del Sistema</b>
	1. El sistema muestra la interfaz de autenticación.
2. El usuario entra su nombre de usuario y contraseña.	3. El sistema comprueba la veracidad de los datos.
	4. El sistema habilita las opciones que ese tipo de usuario puede realizar.


Curso Alternativo de los Eventos	
Acción 2	2.1 Si la información dada por el usuario es incorrecta el sistema muestra un mensaje de error y se regresa al paso 2.
Prioridad	Crítico.
Prototipo de Interfaz.	
	

Tabla 4: Descripción del Caso de Uso Autenticar Usuario.

### 3.4.2.2 Subsistema Gestor de Documentos.

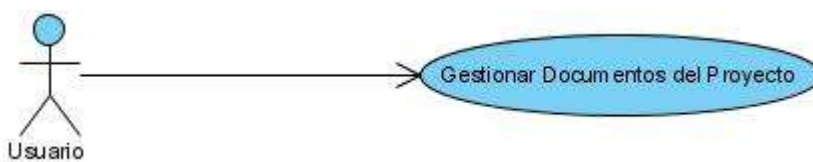


Figura 11: Diagrama de Casos de Uso Subsistema Gestor de Documentos.

Nombre del CU	Gestionar Documentos del Proyecto
Actores	Usuario(inicia)
Objetivo	Manejar la información de los documentos.
Resumen	El caso de uso brinda la posibilidad de insertar, actualizar, ver la traza de un documento, así como listar los documentos existentes, en todas las áreas de la

	aplicación la aplicación para las que tienen permiso.
<b>Referencias</b>	<b>R1</b>
<b>Precondiciones</b>	El usuario debe estar previamente autenticado.
<b>Poscondiciones</b>	
<b>Curso Normal de los Eventos</b>	
<b>Acciones del Actor</b>	<b>Respuesta del Sistema</b>
1. El usuario selecciona la opción “Gestionar Documentos” del Menú.	2. El sistema muestra la interfaz con todas las áreas a cuyos documentos puede acceder.
2.1 Gestionar Documentos del Tiempo. Ver Sección 1.	
2.2 Gestionar Documentos del Costo. Ver Sección 1.	
2.3 Gestionar Documentos de los Riesgos. Ver Sección 1.	
2.4 Gestionar Documentos del Alcance. Ver Sección 1.	
2.5 Gestionar Documentos de la Calidad. Ver Sección 1.	
2.6 Gestionar Documentos de los Recursos Humanos. Ver Sección 1.	
2.7 Gestionar Documentos de las Comunicaciones. Ver Sección 1. Ver Sección 1.	
2.8 Gestionar Documentos del Aprovisionamiento.	

<b>Sección 1</b>	
<b>Acciones del Actor</b>	<b>Respuesta del Sistema</b>
1. El usuario selecciona la opción Gestionar Documentos de un área determinada del menú.	2. El sistema muestra la interfaz con la lista de las últimas actualizaciones de los documentos del área seleccionada, cada uno como un vínculo para descargar el documento y todas las opciones que puede realizar.
2.1 Insertar Nuevo Documento. Ver Sección 1.1	
2.2 Insertar Actualización de un Documento. Ver Sección 1.2	
2.3 Ver traza de un Documento. Ver Sección 1.3	
2.4 Descargar Documento. Ver Sección 1.4.	
<b>Sección 1.1 Insertar Nuevo Documento.</b>	
3. El usuario el usuario selecciona la opción de insertar nuevo documento.	4. El sistema muestra la interfaz para agregar un nuevo documento.
5. El usuario entra el camino del documento que va a insertar y selecciona la opción "Aceptar".	6. El sistema valida los datos entrados.
	7. El sistema almacena el documento, la fecha y la hora en que se guardó, el usuario que lo guardó y el área a la que pertenece el documento y se retorna al paso 2.
<b>Curso Alterno de los Eventos</b>	
Acción 5.1. El usuario selecciona la opción "Cancelar".	6.1 Se retorna al paso 2.

Acción 6.	6.2 El sistema detecta un error en los datos y regresa al paso 4.
<b>Sección 1.2 Insertar Actualización de un Documento.</b>	
3. El usuario selecciona la opción "Actualizar Documento".	4. El sistema muestra la interfaz para actualizar un documento.
5. El usuario entra el camino donde se encuentra la actualización del documento y selecciona "Aceptar".	6. El sistema valida los datos.
	7. El sistema almacena el documento, la fecha y la hora en que se guardó, el usuario que lo guardó y el área a la que pertenece el documento y se retorna al paso 2.
<b>Curso Alterno de los Eventos.</b>	
Acción 5.1 El usuario selecciona la opción cancelar.	6.1 El sistema regresa al paso 2.
Acción 6.	6.2 El sistema detecta un error en los datos y regresa al paso 4.
<b>Sección 1.3 Mostrar la Traza de un Documento.</b>	
3. El usuario selecciona la opción "Mostrar Traza".	4. El sistema muestra la interfaz con las actualizaciones previas del documento marcadas como vínculo para descargar, sus fechas, horas y usuarios que realizaron las actualizaciones del documento.
5 El usuario selecciona la opción "Volver".	6. El sistema regresa al paso 2.
<b>Curso Alternativo de los Eventos.</b>	
Acción 5.1 El usuario da click sobre la actualización que quiere descargar.	Acción 6.2 El sistema levanta la ventana de descarga directa.



	7. Se regresa al paso 4.
<b>Sección 1.4 Descargar Documento.</b>	
3. El usuario da click sobre el documento marcado como vínculo.	4. El sistema levanta la ventana de descarga directa.
	5. Se regresa al paso 2.
<b>Prioridad</b>	Crítico.
<b>Otros</b>	Se puede notar que sea cual sea la selección de la opción de área del documento a gestionar el sistema seguirá los mismos tiempos, la diferencia radica en que cada usuario solo podrá gestionar los documentos de su área a excepción del supervisor que podrá acceder a todas las áreas.
<b>Prototipos de Interfaz.</b>	
<p>Gestionar Documentos:</p> <p><a href="#">del Alcance.</a></p> <p><a href="#">del Costo.</a></p> <p><a href="#">del Tiempo.</a></p> <p><a href="#">de la Calidad.</a></p> <p><a href="#">de los Riesgos.</a></p> <p><a href="#">de las Comunicaciones.</a></p> <p><a href="#">de los Recursos Humanos.</a></p> <p><a href="#">del Aprovisionamiento.</a></p>	

## Gestionar Documentos

[Insertar Nuevo Documento](#)

<a href="#">Documento 1</a>	Actualizar	Ver Traza
<a href="#">Documento 2</a>	Actualizar	Ver Traza
<a href="#">Documento 3</a>	Actualizar	Ver Traza
<a href="#">Documento 4</a>	Actualizar	Ver Traza
<a href="#">Documento 5</a>	Actualizar	Ver Traza

## Insertar Nuevo Documento

## Actualizar Documento

 Examinar

### Traza del Documento.

Nombre	Fecha	Hora	Usuario
<a href="#">Documento 1</a>	14/05/2008	09:30	Jose Francisco Hidalgo
<a href="#">Documento 1</a>	11/07/2008	07:27	Mercedes Remón Lopez
<a href="#">Documento 1</a>	01/10/2008	13:30	Jose Francisco Hidalgo
<a href="#">Documento 1</a>	14/11/2008	23:15	Jose Raul Linares del Castillo
<a href="#">Documento 1</a>	22/11/2008	11:59	Jose Francisco Hidalgo

Tabla 5: Descripción del CUS Gestionar Documentos.

### 3.4.2.3 Subsistema Procesos.

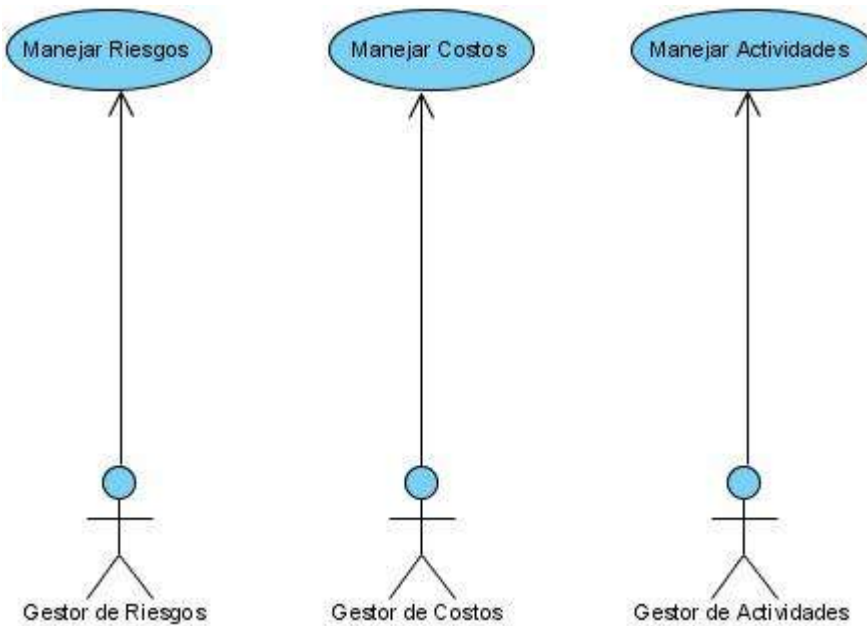


Figura 12: Diagrama de Casos de Uso Subsistema Procesos.

Nombre del CU	Manejar Actividades
Actores	Gestor de Actividades (inicia).
Objetivo	Manejar la información de las actividades.
Resumen	El caso de uso brinda la posibilidad de insertar, modificar y eliminar actividades, así como visualizar el Schedule del proyecto.
Referencias	<b>R2</b>
Precondiciones	El usuario debe estar previamente autenticado.
Poscondiciones	
<b>Curso Normal de los Eventos</b>	
Acciones del Actor	Respuesta del Sistema
1. El Gestor de Actividades selecciona la opción "Gestionar Actividades" del menú.	2. El sistema muestra la interfaz una lista de las actividades ya creadas y todas las opciones que puede seleccionar.
2.1 Insertar Nueva Actividad. Ver	

Sección 1.	
2.2 Modificar Actividad. Ver Sección 2.	
2.3 Eliminar Actividad. Ver Sección 3.	
2.4 Mostrar el Schedule. Ver Sección 4.	
2.5 Mostrar Actividad. Ver Sección 5.	
<b>Sección 1. Insertar Nueva Actividad.</b>	
3. El usuario el usuario selecciona la opción de insertar nueva actividad.	4. El sistema muestra la interfaz para agregar una nueva actividad.
5. El usuario entra el nombre de la actividad, la fecha de inicio y fin, el responsable de la actividad, el código de la actividad, la precedencia y selecciona la opción "Aceptar".	6. El sistema valida los datos entrados (debe haber insertado al menos 1 recurso).
5.1 El usuario selecciona la opción "Recursos". Ver Sección 1.1	
	7. El sistema almacena los datos y se regresa al paso 2.
<b>Curso Alternativo de los Eventos</b>	
Acción 5.1. El usuario selecciona la opción "Cancelar".	6.1 Se retorna al paso 2.
Acción 6.	6.2 El sistema detecta un error en los datos y regresa al paso 4.

<b>Sección 1.1</b>	
<b>Acciones del Actor</b>	<b>Respuesta del Sistema</b>
5.1 El equipo de gestión del tiempo selecciona la opción "Recursos" del menú.	6.1 El sistema muestra la interfaz una lista de los recursos ya creados y todas las opciones que puede seleccionar.
7.1 Insertar Nuevo Recurso. Ver Sección 1.1.1	
7.2 Modificar Recurso. Ver Sección 1.1.2	
7.3 Eliminar Recurso. Ver Sección 1.1.3	
7.4 Regresar. Ver Sección 4.	
<b>Sección 1.1.1 Insertar Nuevo Recurso.</b>	
8. El usuario el usuario selecciona la opción de insertar nuevo recurso.	9. El sistema muestra la interfaz para agregar un nuevo recurso.
10. El usuario entra el nombre del recurso, la unidad de medición (Kg, m <sup>3</sup> , m <sup>2</sup> , m, unidades), la cantidad necesitada del recurso y selecciona la opción "Aceptar".	11. El sistema valida los datos entrados.
	12. El sistema guarda los datos y regresa al paso 5.
<b>Curso Alternativo de los Eventos</b>	
Acción 10.1 El usuario selecciona la opción "Cancelar".	11.1 Se retorna al paso 5.
Acción 11.	11.2 El sistema detecta un error en los datos y regresa al paso 8.

<b>Sección 1.1.2 Modificar Recurso.</b>	
8. El usuario selecciona la opción “Modificar Recurso”.	9. El sistema muestra la interfaz para modificar un recurso.
10. El usuario modifica los datos y selecciona “Aceptar”.	11. El sistema valida los datos.
	12. El sistema guarda los datos y retorna al paso 2.
<b>Curso Alternativo de los Eventos.</b>	
Acción 10.1 El usuario selecciona la opción cancelar.	11.1 El sistema regresa al paso 5.
Acción 11.	11.2 El sistema detecta un error en los datos y regresa al paso 8.
<b>Sección 1.1.3 Eliminar Recurso.</b>	
8. El usuario selecciona la opción “Eliminar Recurso”.	9. El sistema pide confirmación para eliminar.
10. El usuario selecciona la opción “Aceptar”.	11. El sistema elimina los datos del recurso y regresa al paso 5.
<b>Curso Alternativo de los Eventos.</b>	
Acción 10.1 El usuario selecciona la opción “Cancelar”	11.1 El sistema regresa al paso 5.
<b>Sección 1.1.4. Regresar</b>	
8. El usuario selecciona la opción “Regresar”	9. El sistema valida la existencia de al menos un recurso y se regresa al paso 5.
<b>Curso Alternativo de los Eventos.</b>	
Acción 8.	9.1 El sistema detecta la falta de al menos un recurso y regresa al paso 8.
<b>Sección 2. Modificar Actividad.</b>	
3. El usuario selecciona la opción	4. El sistema muestra la interfaz para

“Modificar Actividad”.	modificar una actividad.
5. El usuario modifica los datos y selecciona “Aceptar”.	6. El sistema valida los datos.
5.1 El usuario selecciona la opción recursos. Ver sección 1.1.	
	7. El sistema guarda los datos y retorna al paso 2.
<b>Curso Alternativo de los Eventos.</b>	
Acción 5. El usuario selecciona la opción cancelar.	6.1 El sistema regresa al paso 2.
Acción 6.	6.3 El sistema detecta un error en los datos y regresa al paso 4.
<b>Sección 3. Eliminar Actividad.</b>	
3. El usuario selecciona la opción “Eliminar Actividad”.	4. El sistema pide confirmación para eliminar.
5 El usuario selecciona la opción “Aceptar”.	6. El sistema elimina los datos de la actividad y regresa al paso 2.
<b>Curso Alternativo de los Eventos.</b>	
Acción 5. El usuario selecciona la opción “Cancelar”	6.1 El sistema regresa al paso 2.
<b>Sección 4. Mostrar Schedule.</b>	
3. El usuario selecciona la opción “Mostrar Schedule”	4. El sistema muestra una interfaz con un diagrama de Gantt mostrando una vista general del proyecto.
5. El usuario selecciona la opción “Regresar”	6. El sistema retorna al paso 2.
<b>Sección 5. Mostrar Actividad</b>	
3. El usuario da click sobre la	4. El sistema muestra una interfaz con una



actividad que quiere mostrar.	tabla mostrando el nombre de la actividad, su fecha de inicio y fin, su responsable y una lista con el nombre de los recursos que se necesitan para ejecutarla y la cantidad.
5. El usuario selecciona la opción "Regresar"	6. El sistema retorna al paso 2.
<b>Prioridad</b>	Crítico.

**Prototipos de Interfaz.**

Manejar Actividades

[Insertar Nueva Actividad](#)

[Mostrar Schedule](#)

<b>Actividad 1</b>	Mostrar	Modificar	Eliminar
<b>Actividad 2</b>	Mostrar	Modificar	Eliminar
<b>Actividad 3</b>	Mostrar	Modificar	Eliminar
<b>Actividad 4</b>	Mostrar	Modificar	Eliminar
<b>Actividad 5</b>	Mostrar	Modificar	Eliminar

Insertar Nueva Actividad

**Nombre**

**Código**

**Fecha de Inicio(dd/mm/aaaa)**

**Fecha de Fin(dd/mm/aaaa)**

**Nombre del Responsable**

**Precedencia**

Actividad 1

Actividad 2

Actividad 3

Actividad 4

**Recursos**

## Gestionar Recursos

[Insertar Nuevo Recurso](#)

[Regresar](#)

<a href="#">Recurso 1</a>	Modificar	Eliminar
<a href="#">Recurso 2</a>	Modificar	Eliminar
<a href="#">Recurso 3</a>	Modificar	Eliminar
<a href="#">Recurso 4</a>	Modificar	Eliminar
<a href="#">Recurso 5</a>	Modificar	Eliminar

## Insertar Nuevo Recurso

Nombre

Unidad de Medición

Cantidad

Aceptar

Cancelar

### Modificar Recurso

**Nombre**

Recurso 1 metros

**Cantidad**

100

Aceptar

Cancelar

### Modificar Actividad

**Nombre**

Actividad 5

**Código**

R05

**Fecha de Inicio(dd/mm/aaaa)**

24/10/2004

**Fecha de Fin(dd/mm/aaaa)**

15/12/2004

**Nombre del Responsable**

Ing. Rodobaldo Hernán

**Precedencia**

- Actividad 1
- Actividad 2
- Actividad 3
- Actividad 4

Recursos

Aceptar

Cancelar

Mostrar Actividad	
Nombre:	Actividad 5
Código:	R05
Fecha de Inicio :	22/10/2004
Fecha de Fin:	15/12/2004
Responsable:	Ing. Rodobaldo Hernandez Perez
Precedencia:	Actividad 1, Actividad 2
Recursos	
<b>Recurso 1</b>	100 metros
<b>Recurso 2</b>	200 unidades
<b>Recurso 3</b>	15 Kg
<b>Recurso 4</b>	400 unidades
<b>Recurso 5</b>	1000 metros
<a href="#">Regresar</a>	

Tabla 6: Descripción del CUS Manejar Actividades.

Nombre del CU	Manejar Costos.
Actores	Gestor de Costos (inicia).
Objetivo	Manejar la información de los Costos.
Resumen	El caso de uso brinda la posibilidad de insertar y modificar los costos así como ver un informe detallado de estos.
Referencias	<b>R3</b>
Precondiciones	El usuario debe estar previamente autenticado.
Poscondiciones	
<b>Curso Normal de los Eventos</b>	
Acciones del Actor	Respuesta del Sistema
1. El Gestor de Costos selecciona la opción "Gestionar Costos" del menú.	2. El sistema muestra la interfaz con una lista de las actividades creadas y las opciones que puede seleccionar.
3. El usuario selecciona la actividad a cuyo recurso quiere	4. El sistema muestra la interfaz con una lista de los recursos que tiene previstos.

asignarle un costo.	
5. El usuario selecciona el recurso para asignarle un costo.	6. El sistema le muestra una interfaz para que entre los datos.
7. El usuario entra el precio unitario del recurso y su proveedor y selecciona "Aceptar"	8. El sistema valida los datos, los almacena y regresa al paso 3.
<b>Curso Alternativo de los Eventos</b>	
Acción 3.1 El usuario selecciona la opción "Mostrar Informe detallado de Costos". Ver Sección 1.	
Acción 5.1 El usuario selecciona la opción "Regresar".	6.1 El sistema regresa al paso 2.
Acción 7.1 El usuario selecciona la opción "Cancelar".	8.1 El sistema regresa al paso 3.
<b>Sección 1. Mostrar Informe detallado de Costos.</b>	
3.1 El usuario selecciona la opción "Mostrar Informe detallado de Costos".	4. El sistema muestra la interfaz con un reporte detallado de las actividades con sus recursos y sus costos, unitarios y totales, respectivamente.
5. El usuario selecciona la opción "Regresar".	6. El sistema regresa al paso 2.
<b>Prioridad</b>	Crítico.
<b>Prototipos de Interfaz.</b>	

## Manejar Costos

[Actividad 1](#)

[Actividad 2](#)

[Actividad 3](#)

[Actividad 4](#)

[Actividad 5](#)

[Mostrar Informe Detallado de Costos](#)

## Manejar Costos

**Actividad 1**

[Recurso 1](#)

Asignar Costo

[Recurso 2](#)

Asignar Costo

[Recurso 3](#)

Asignar Costo

[Recurso 4](#)

Asignar Costo

[Recurso 5](#)

Asignar Costo

[Regresar](#)

## Manejar Costos

Recurso 1

Precio Unitario

Aceptar

Cancelar

### Informe Detallado de Costos

	Nombre del Recurso	Precio Unitario	Precio Total
Actividad 1	Recurso 1	100.05	10005.00
	Recurso 2	20.00	400.00
Actividad 2	Recurso 3	50.07	350.49
Actividad 3	Recurso 4	90.00	900000.00
	Recurso 5	15.35	30.70
	Recurso 6	5.07	35.49
<b>Total</b>			<b>910821.68</b>

[Regresar](#)

Tabla 7: Descripción del CUS Manejar Costos.

Nombre del CU	Manejar Riesgos.
Actores	Gestor de Riesgos (inicia).
Objetivo	Manejar la información de los Riesgos.
Resumen	El caso de uso brinda la posibilidad de insertar, modificar, eliminar y ejecutar los riesgos.
Referencias	R4

<b>Precondiciones</b>	El usuario debe estar previamente autenticado.
<b>Poscondiciones</b>	
<b>Curso Normal de los Eventos</b>	
<b>Acciones del Actor</b>	<b>Respuesta del Sistema</b>
1. El Gestor de Riesgos selecciona la opción “Gestionar Riesgos” del menú.	2. El sistema muestra la interfaz con una lista de los riesgos ya creados y las opciones que puede seleccionar.
2.1 Insertar Nuevo Riesgo. Sección 1.	
2.2 Modificar Riesgo. Sección 2.	
2.3 Eliminar Riesgo. Sección 3.	
2.4 Mostrar Riesgo. Sección 4.	
<b>Sección 1. Insertar Nuevo Riesgo.</b>	
3. El usuario selecciona la opción “Insertar Nuevo Riesgo”.	4. El sistema muestra una interfaz para insertar un riesgo nuevo.
5. El usuario entra el nombre del riesgo, el código del riesgo, el nombre del responsable de darle respuesta, el índice de probabilidad, el índice de impacto, una descripción del riesgo y selecciona la opción “Aceptar”.	6. El sistema valida los datos.
	7. El sistema almacena los datos y regresa al paso 2.
<b>Curso Alternativo de los Eventos.</b>	
Acción 5.1 El usuario selecciona la opción “Cancelar”.	6.1 El sistema regresa al paso 2.



Acción 6.	6.2 El sistema detecta un error en los datos y regresa al paso 4.
<b>Sección 2. Modificar Riesgo.</b>	
3. El usuario selecciona la opción "Modificar Riesgo"	4. El sistema muestra la interfaz para modificar los datos del Riesgo.
5. El usuario modifica los valores que desea y selecciona "Aceptar".	6. El sistema valida los datos.
	7. El sistema guarda los datos en la base de datos.
<b>Curso Alternativo de los Eventos.</b>	
Acción 5.1 El usuario selecciona la opción "Cancelar".	6.1 El sistema regresa al paso 2.
Acción 6.	6.2 El sistema detecta un error en los datos y regresa al paso 4.
<b>Sección 3. Eliminar Riesgo.</b>	
3. El usuario selecciona la opción "Eliminar Riesgo".	4. El sistema pide una confirmación para eliminar el riesgo seleccionado.
5. El usuario selecciona la opción "Aceptar".	6. El sistema elimina los datos del riesgo y regresa al paso 2.
<b>Curso Alternativo de los Eventos.</b>	
Acción 5.1 El usuario selecciona la opción "Cancelar".	6.1 El sistema regresa al paso 2.
<b>Sección 4. Mostrar Riesgo</b>	
3. El usuario da click sobre el riesgo que quiere mostrar.	4. El sistema muestra una interfaz con una tabla mostrando el nombre del riesgo, su código, su índice de probabilidad, su índice de impacto, su índice de severidad

	(probabilidad * impacto), el responsable del plan de respuesta y su descripción.
5. El usuario selecciona la opción "Regresar".	6. El sistema regresa al paso 2.
<b>Prioridad</b>	Crítico.

**Prototipos de Interfaz.**

Manejar Riesgos

[Insertar Nuevo Riesgo](#)

<b>Riesgo 1</b>	Mostrar	Modificar	Eliminar
<b>Riesgo 2</b>	Mostrar	Modificar	Eliminar
<b>Riesgo 3</b>	Mostrar	Modificar	Eliminar
<b>Riesgo 4</b>	Mostrar	Modificar	Eliminar
<b>Riesgo 5</b>	Mostrar	Modificar	Eliminar

Insertar Nuevo Riesgo

Nombre  Código

Indice de Probabilidad  Indice de Impacto

Nombre del Responsable de la Respuesta

Descripción

### Modificar Riesgo

**Nombre**  **Código**

**Índice de Probabilidad**  **Índice de Impacto**

**Nombre del Responsable de la Respuesta**

**Descripción**

Texto Texto Texto Texto Texto Texto Texto Texto

Texto Texto Texto Texto Texto Texto Texto Texto

Texto Texto Texto Texto Texto Texto Texto Texto

Texto Texto Texto Texto Texto Texto Texto Texto

Texto Texto Texto Texto Texto Texto Texto Texto

Texto Texto Texto Texto Texto Texto Texto Texto

Texto Texto Texto Texto Texto Texto Texto Texto

Texto Texto Texto Texto Texto Texto Texto Texto

---

### Mostrar Riesgo

**Nombre:** Déficit de Cemento

**Código:** T20

**Índice de Probabilidad:** 0.3

**Índice de Impacto:** 0.5

**Índice de Severidad :** 0.15

**Responsable de la Respuesta:** Lic. Maria Gé Suarez

**Descripción:**

Texto Texto Texto Texto Texto Texto Texto Texto

Texto Texto Texto Texto Texto Texto Texto Texto

Texto Texto Texto Texto Texto Texto Texto Texto

Texto Texto Texto Texto Texto Texto Texto Texto

Texto Texto Texto Texto Texto Texto Texto Texto

Texto Texto Texto Texto Texto Texto Texto Texto

Texto Texto Texto Texto Texto Texto Texto Texto

Texto Texto Texto Texto Texto Texto Texto Texto

Texto Texto Texto Texto Texto Texto Texto Texto

Texto Texto Texto Texto Texto Texto Texto Texto

Texto Texto Texto Texto Texto Texto Texto Texto

Texto Texto Texto Texto Texto Texto Texto Texto

Texto Texto Texto Texto

[Regresar](#)

**Tabla 8: Descripción del CUS Manejar Riesgos.**

### **3.5 Conclusiones.**

En este capítulo se describieron los diferentes procesos del negocio relacionados con la Gestión de Proyectos, se describieron los actores y trabajadores del negocio, así como los casos de usos del negocio. Se elaboró un diagrama de casos de uso del negocio para ayudar a una mejor comprensión del problema a solucionar. Además se inició el desarrollo de la propuesta de solución, basándose en los procesos del negocio se definieron las funcionalidades que el sistema debe tener así como el prototipo de interfaz que debe tener el sistema para comunicarse con los usuarios.

# CAPÍTULO 4: Construcción de la solución propuesta.

## 4.1 Introducción

En este capítulo se realiza la construcción de la solución propuesta en el capítulo anterior, se realiza la definición de las clases que serán utilizadas y se confecciona el diseño de la base de datos.

## 4.2 Modelo del Análisis.

### 4.2.1 Diagrama de Clases del Análisis.

#### 4.2.1.1 Subsistema Autenticación.

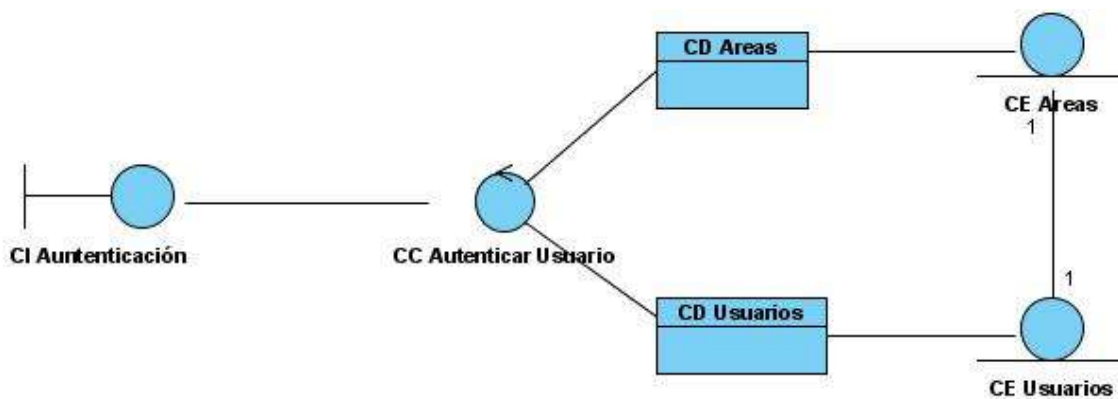


Figura 13: Diagrama de Clases del Análisis CU Autenticar Usuario.

#### 4.2.1.2 Subsistema Gestor de Documentos.

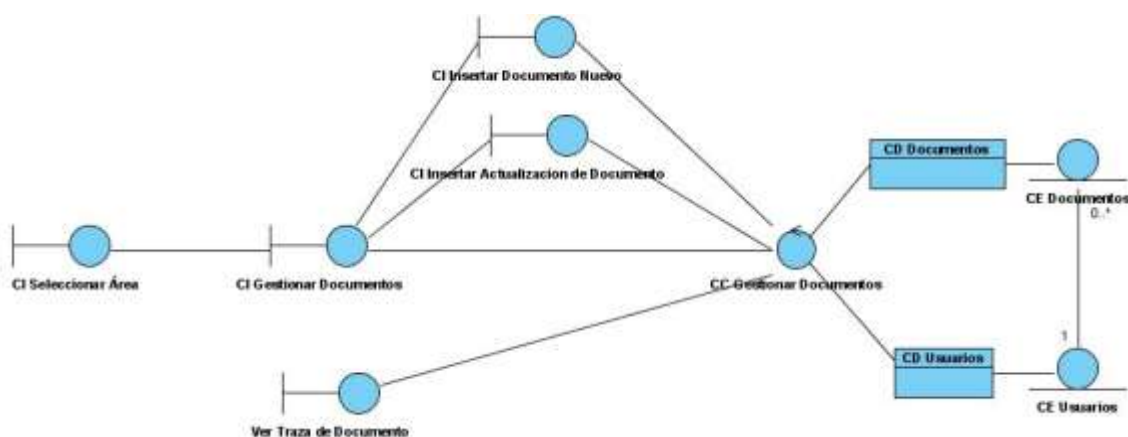


Figura 14: Diagrama de Clases del Análisis CU Gestionar Documentos.

### 4.2.1.3 Subsistema Procesos.

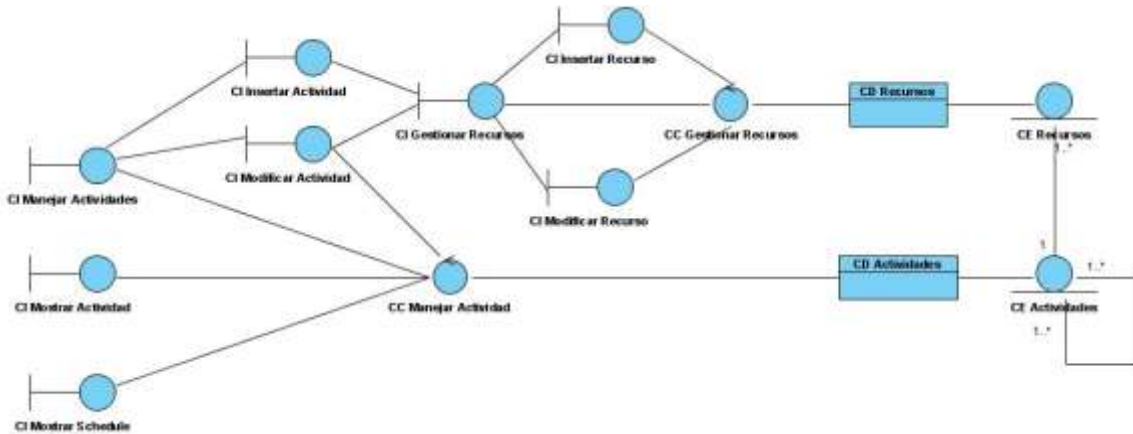


Figura 15: Diagrama de Clases del Análisis CU Manejar Actividades.

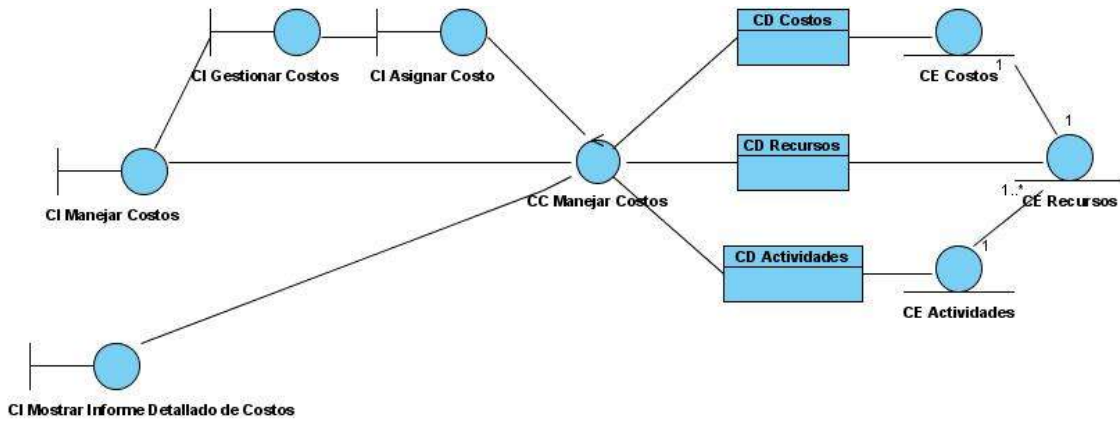


Figura 16: Diagrama de Clases del Análisis CU Manejar Costos.

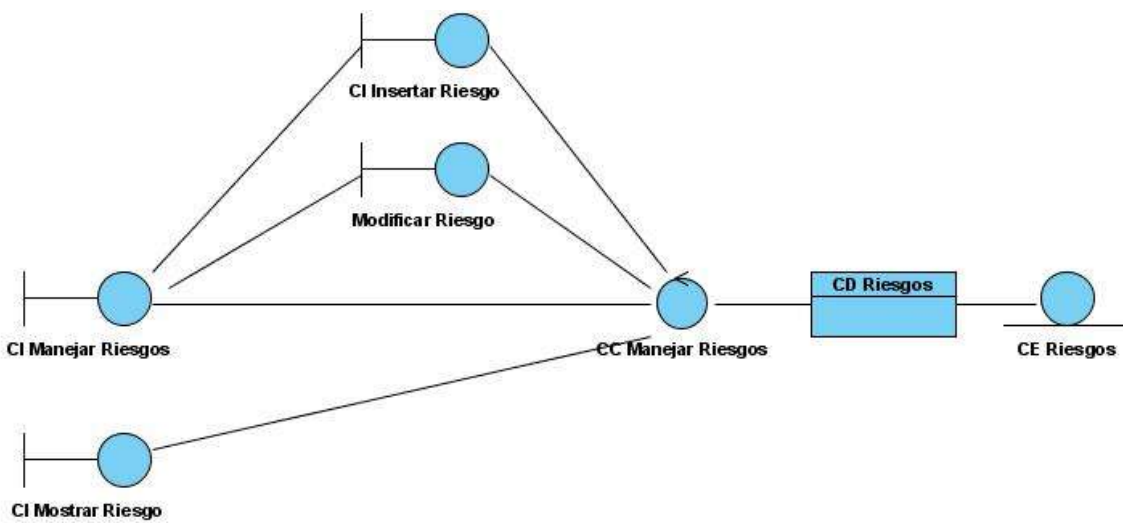


Figura 17: Diagrama de Clases del Análisis CU Manejar Riesgos.

### 4.3 Modelo del Diseño.

#### 4.3.1 Diagramas de Clases del Diseño.

##### 4.3.1.1 Subsistema Autenticación.

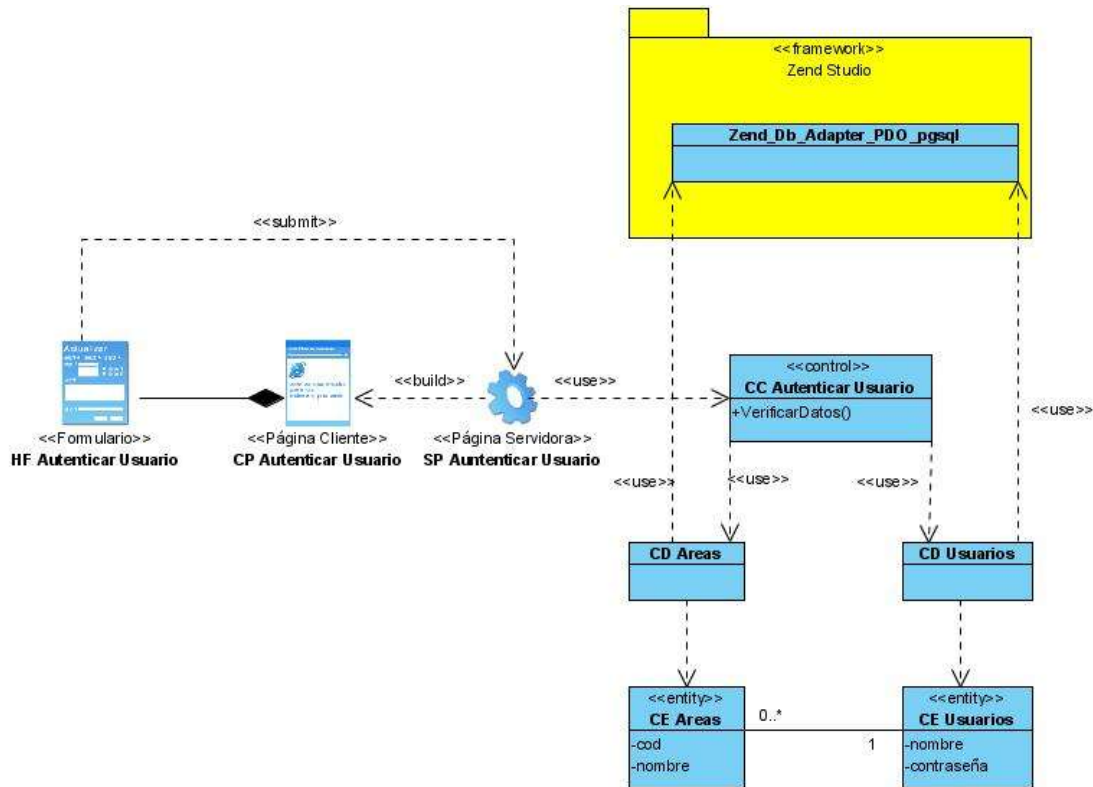


Figura 18: Diagrama de Clases del Diseño CU Autenticar Usuario.

### 4.3.1.2 Subsistema Gestor de Documentos.

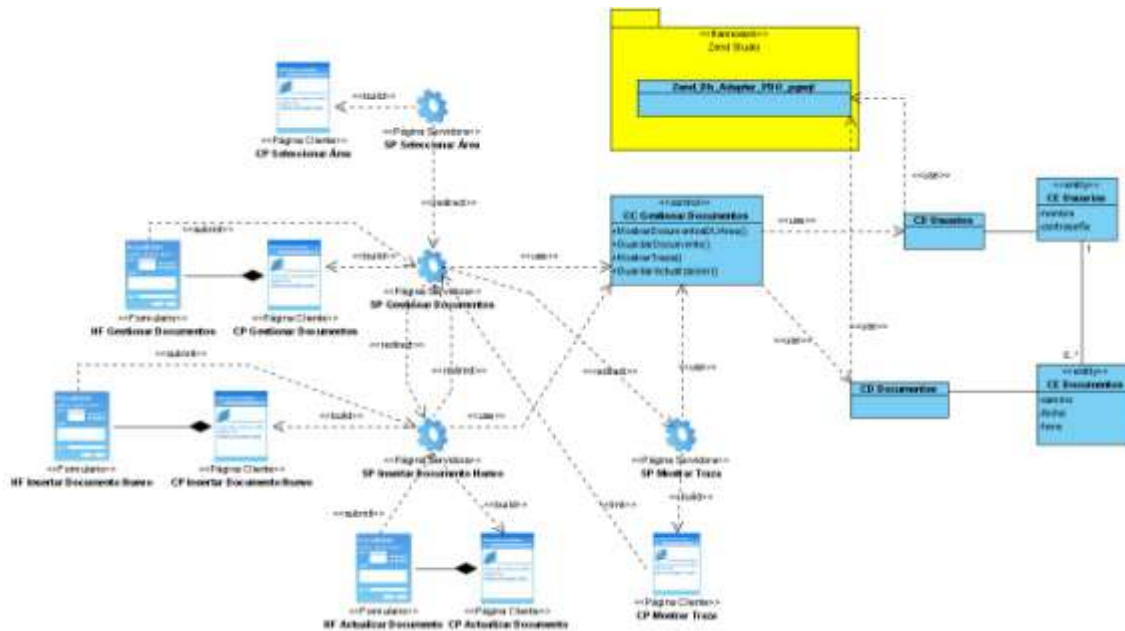


Figura 19: Diagrama de Clases del Diseño CU Gestionar Documentos.

### 4.3.1.3 Subsistema Procesos.

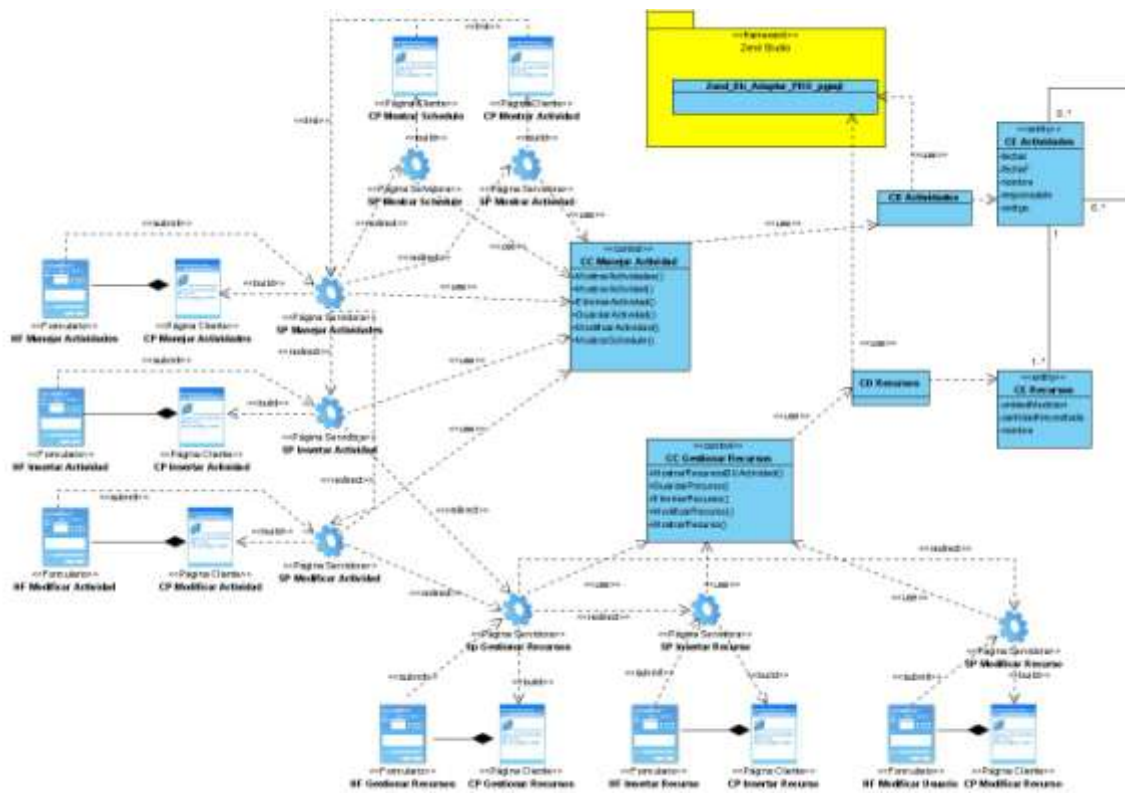


Figura 20: Diagrama de Clases del Diseño CU Manejar Actividades.





## 4.4 Modelo de Datos.

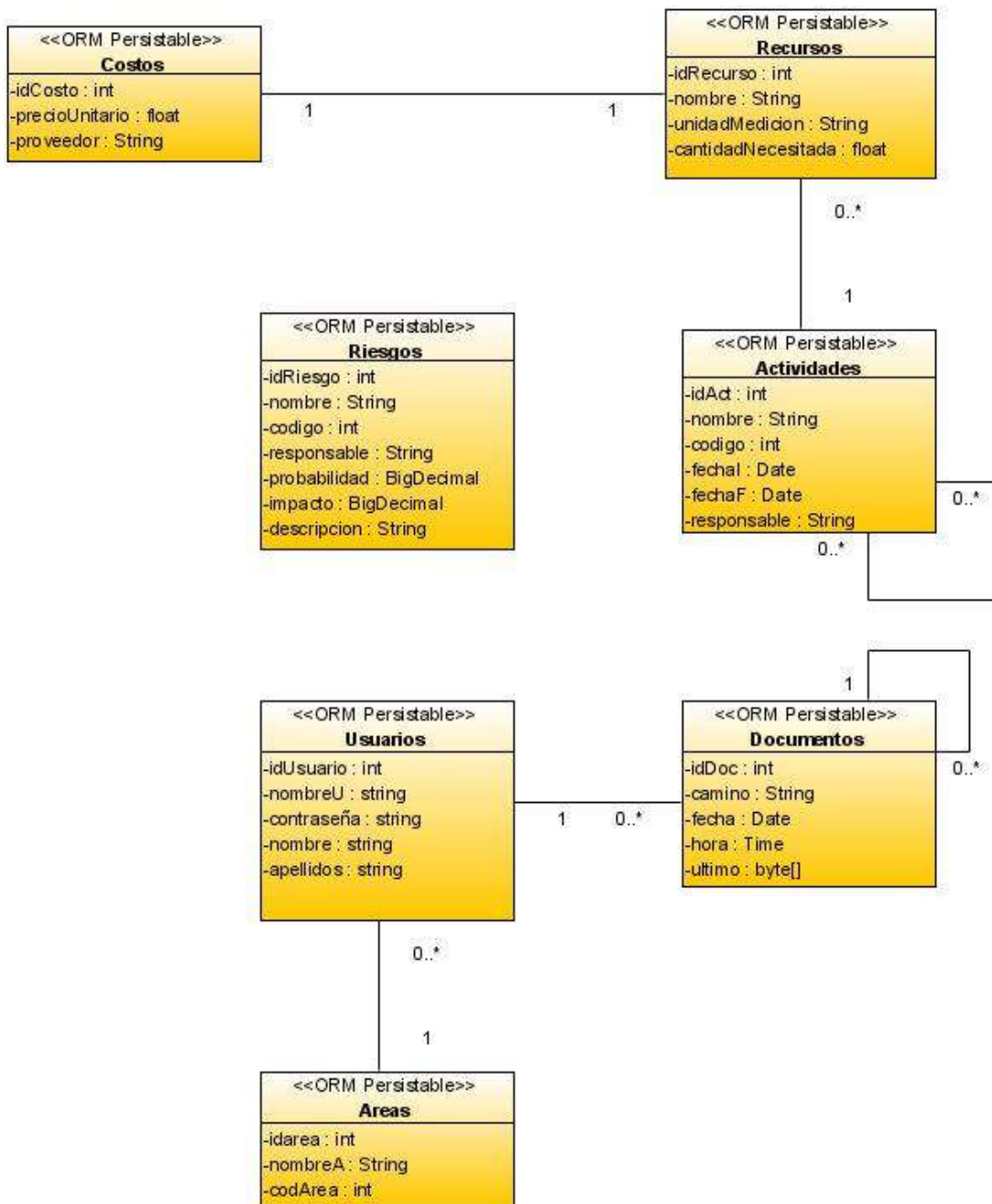


Figura 23: Diagrama de Clases Persistentes.

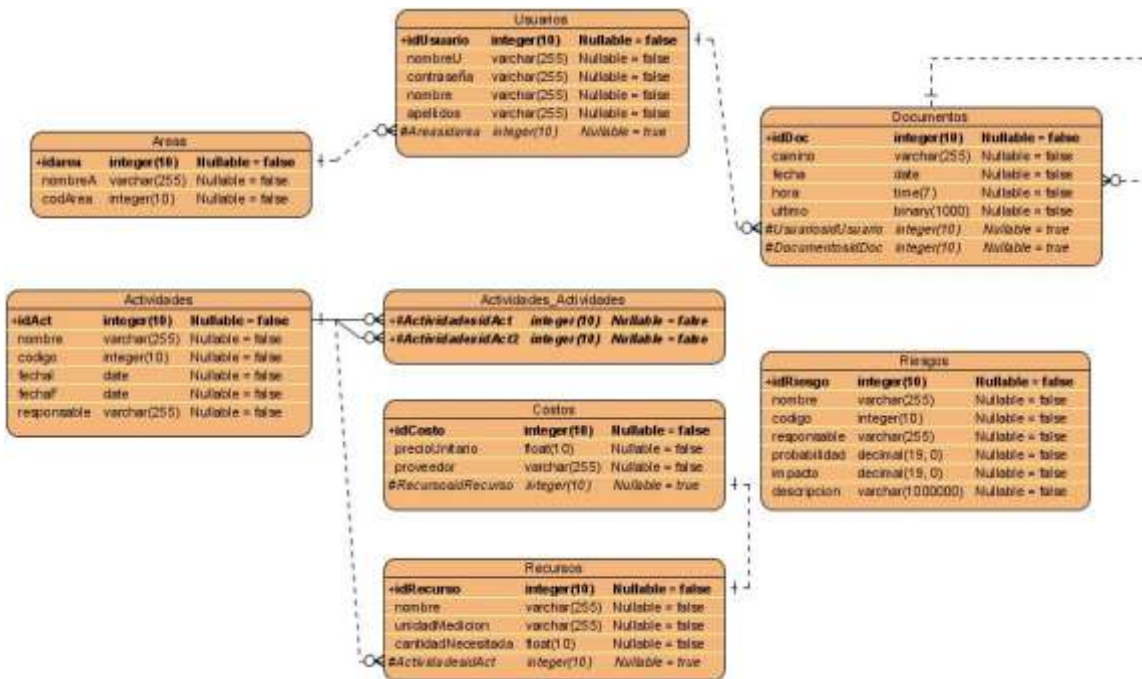


Figura 24: Modelo de Datos.

## 4.5 Conclusiones.

En este capítulo se realizó la construcción de la solución previamente propuesta hasta el flujo de trabajo de Análisis y Diseño. Se definieron las clases a implementar para dar funcionalidad a los casos de uso y se diseñó la estructura de tablas y relaciones de la base de datos.

## Conclusiones

Se realizó un estudio profundo acerca de la Gestión de Proyectos como disciplina, también se profundizó en las tendencias y tecnologías que más se usan actualmente a nivel mundial, y se propuso una solución al problema de la inexistencia en software libre de herramientas que informaticen gran parte de los procesos de la Gestión de Proyectos.

Se le da cumplimiento al objetivo planteado de diseñar una herramienta en software libre que informatice gran parte de los procesos de la Gestión de Proyectos y en especial los vinculados a la Gestión de la Calidad, el Aprovisionamiento y los Riesgos.

Finalmente se concluye que:

- Se cumplieron las tareas previstas para el cumplimiento del objetivo del trabajo.
- .El uso de los métodos de investigación facilitó el estudio profundo del objeto de estudio.
- La propuesta de herramienta se ha diseñado según las teorías del PMI de lo que es una “buena práctica” en la Gestión de Proyectos.
- La aplicación diseñada favorecerá el proceso de toma de decisiones, al ser una ayuda significativa a la disponibilidad de la información.
- La herramienta servirá de apoyo a la Gestión de Proyectos, informatizando gran parte de los procesos de la disciplina y específicamente los relacionados con la Gestión de la Calidad, el Aprovisionamiento y los Riesgos.

## Recomendaciones

1. Continuar el ciclo de desarrollo de la herramienta, realizando nuevas iteraciones en cada uno de los flujos de trabajo, para obtener una herramienta competitiva en el mercado del software.
2. Investigar a profundidad cada una de las Áreas de Conocimientos de la Gestión de Proyecto con el objetivo de descubrir otros procesos informatizables.
3. Agregar a la herramienta la posibilidad de gestionar varios proyectos a la vez e investigar el espectro de procesos informatizables que esta funcionalidad permitiría.
4. Desarrollar una versión de la herramienta para escritorio, para que los proyectos pequeños que puedan ser gestionados por una sola persona no necesiten adquirir computadoras extra para servidores y clientes.

## Bibliografía Consultada

- Alanis, Macedonio. 2005. *Planeación de Recursos Empresariales*. Monterrey : s.n., 2005.
- **Desarrollo Web, Tu mejor ayuda para aprender a hacer webs.** [En línea] [Citado el: 24 de abril de 2008.] [www.desarrolloweb.com](http://www.desarrolloweb.com).
- *El concepto y alcance de la gestión tecnológica.* Restrepo Gonzalez, Guillermo. 2000. 21, Medellín : Revista Facultad de Ingeniería, 2000.
- Factoría de Internet. 2003. **El Portal del Webmaster.** [En línea] Factoría de Internet, 2003. [Citado el: 24 de abril de 2008.] [www.webtaller.com](http://www.webtaller.com).
- 2001. **Free Software Project Management Tutorial and Course.** [En línea] Project Shrink Publishing, 2001. [Citado el: 23 de abril de 2008.] <http://www.softwareprojects.org/>.
- Harper, Mitchell. **What is ASP.NET?** [En línea] [Citado el: 25 de abril de 2008.] [www.javascriptkit.com](http://www.javascriptkit.com).  
— . 1998. **Your First SQL Server 2000 Database .** [En línea] SitePoint Pty. Ltd., 1998. [Citado el: 23 de abril de 2008.] [www.sitepoint.com](http://www.sitepoint.com).
- Inc., Project Management Institute. 2004. *A Guide to the Project Management Body of Knowledge*. s.l. : (PMBOK® Guide), 2004. 1-930699-73-5.
- **Increase productivity, communication, and collaboration using UML visual modeling platform.** [En línea] [Citado el: 23 de abril de 2008.] <http://www.visual-paradigm.com/>.
- 2007. **Kennesaw State University.** [En línea] Kennesaw State University, 2007. [Citado el: 21 de abril de 2008.] <http://atlas.kennesaw.edu/>.
- Masip, David. **Que es Oracle?** [En línea] [Citado el: 20 de abril de 2008.] [www.desarrolloweb.com](http://www.desarrolloweb.com).
- **Microsoft Corporation.** [En línea] Microsoft Corporation. [Citado el: 27 de abril de 2008.] [www.microsoft.com](http://www.microsoft.com).
- 2008. **MSDN: Microsoft Developer Network.** [En línea] Microsoft Corporation, 2008. [Citado el: 30 de abril de 2008.] <http://msdn.microsoft.com/en-us/default.aspx>.

- **Netveloper - ASP.NET, C#, VB.NET, ADO.NET, .NET Framework, SQL Server, Windows Server System.** [En línea] [Citado el: 21 de abril de 2008.] <http://www.netveloper.com/>.
- Pecos Martinez, Daniel. **PostgreSQL vs. MySQL.** [En línea] [Citado el: 22 de abril de 2008.] [www.netpecos.org](http://www.netpecos.org).
- **pgAdmin III: PostgreSQL administration and management tools.** [En línea] [Citado el: 24 de abril de 2008.] <http://www.pgadmin.org>.
- Rumbaugh, James, Jacobson, Ivar y Booch, Grady. ***El Lenguaje Unificado de Modelado. Manual de Referencia.*** s.l. : Addison Wesley.
- 2008. **SOA, Middleware, Integration Software, Financial Messaging, Open Source, CORBA.** [En línea] Iona Technologies, 2008. [Citado el: 29 de abril de 2008.] [www.iona.com](http://www.iona.com).
- 2004. **Startvbdotnet: : Free VB .NET, ASP.NET Tutorials, Code, Concepts.** [En línea] 2004. [Citado el: 25 de abril de 2008.] <http://www.startvbdotnet.com/>.
- Tagliavini, M., y otros. 2002. ***Exploring the use of ERP systems by SMEs.*** Orlando, FI : 6th World Multiconference on Systemics, 2002.
- 2001. **Web Wiz.** [En línea] 2001. [Citado el: 28 de abril de 2008.] <http://www.webwizguide.com/>.
- 2001. **Website Hosting - Free Domain, Live support, Sitebuilder.** [En línea] Flux Services, 2001. [Citado el: 23 de abril de 2008.] <http://www.ultrawebsitehosting.com/>.
- Wells, Don. 2006. **Extreme Programming: A gentle introduction.** [En línea] 17 de Febrero de 2006. [Citado el: 27 de abril de 2008.] [www.extremeprogramming.org](http://www.extremeprogramming.org).
- Zend Technologies Ltd. 2008. **PHP Development & Production Software - PHP Tutorials Training & Certification - Learn PHP - Zend.com.** [En línea] Zend Technologies Ltd., 2008. [Citado el: 24 de abril de 2008.] <http://www.zend.com>.

## Citas Bibliográficas

1. Alanis, Macedonio. *Planeación de Recursos Empresariales*. Monterrey : s.n., 2005.
2. Tagliavini, M., et al. *Exploring the use of ERP systems by SMEs*. Orlando, FL : 6th World Multiconference on Systemics, 2002.
3. Project Management Institute. [Online] [Cited: Abril 17, 2008.]  
<http://www.pmi.org>.
4. Inc., Project Management Institute. *A Guide to the Project Management Body of Knowledge*. s.l. : (PMBOK® Guide), 2004. 1-930699-73-5.
5. *El concepto y alcance de la gestión tecnológica*. Restrepo Gonzalez, Guillermo. 21, Medellín : Revista Facultad de Ingeniería, 2000.



## Glosario de Siglas y Términos

- WBS: Work Breakdown Structure. Estructura de Desglose del Trabajo en español. Es una estructura exhaustiva, jerárquica y descendente formada por los entregables y las tareas necesarias para completar un proyecto.
- Stakeholders. Se refiere a quienes pueden afectar o son afectados por las actividades de una empresa.
- RUP: Rational Unified Process, Proceso Unificado de Desarrollo de Software en español.
- UML: Lenguaje Unificado de Modelado, usado para modelar sistemas de software.
- PHP: Hypertext Pre-Processor. Es un ambiente script del lado del servidor que permite crear y ejecutar aplicaciones Web dinámicas e interactivas.
- SQL: Structured Query Language. Es un lenguaje declarativo de acceso a bases de datos que permite especificar diversos tipos de operaciones sobre las mismas. Aúna características del álgebra y el cálculo relacional permitiendo lanzar consultas con el fin de recuperar información de interés de una base de datos.
- Diagrama de Gantt: Un diagrama horizontal especial que proporciona una descripción gráfica y una programación cronometrada de todas las actividades, elementos y dependencias de un proyecto o programa.
- Schedule: Plan, programa, proyecto, calendario.
- Documentos del Proyecto: Se refiere a toda la documentación generada por los procesos de la Gestión de Proyectos, por ejemplo, los Documentos del Tiempo son todos los documentos generados por los procesos que se ejecutan en la Gestión del Tiempo del Proyecto. Lo mismo pasa para las demás áreas de Gestión como la del Alcance, los Costos, los Riesgos, la Calidad, el Aprovisionamiento, los Recursos Humanos y las Comunicaciones.